PÓS-GRADUAÇÃO EM GEOCIÊNCIAS APLICADAS E GEODINÂMICA

INSTITUTO DE GEOCIÊNCIAS

UNIVERSIDADE DE BRASÍLIA

Dissertação de Mestrado nº 192

EXTRAÇÃO AUTOMÁTICA DE EDIFICAÇÕES PARA A PRODUÇÃO CARTOGRÁFICA UTILIZANDO INTELIGÊNCIA ARTIFICIAL

PHILIPE BORBA

Orientador: Prof. Dr. Edilson de Souza Bias

Coorientador: Prof. Dr. Nilton Correia da Silva

Brasília, Fevereiro de 2022

PÓS-GRADUAÇÃO EM GEOCIÊNCIAS APLICADAS E GEODINÂMICA
INSTITUTO DE GEOCIÊNCIAS
UNIVERSIDADE DE BRASÍLIA

EXTRAÇÃO AUTOMÁTICA DE EDIFICAÇÕES PARA A PRODUÇÃO CARTOGRÁFICA UTILIZANDO INTELIGÊNCIA ARTIFICIAL

Área de Concentração: Geoprocessamento e Análise Ambiental
Linha de Pesquisa: Avaliação de Dados e Técnicas de Sensoriamento Remoto, Geoprocessamento, Cartografia e Geodésia

Philipe Borba

Dissertação de Mestrado submetida ao Instituto de Geociências da Universidade de Brasília, como parte dos requisitos para obtenção do grau de Mestre em Geociências Aplicadas.

Orientador: Prof. Dr. Edilson de Souza Bias
Coorientador: Prof. Dr. Nilton Correia da Silva

Brasília - DF
Fevereiro de 2022

Ficha catalográfica elaborada automaticamente,
com os dados fornecidos pelo(a) autor(a)

```
         Borba, Philipe
BB726e      Extração Automática de Edificações para a Produção
         Cartográfica Utilizando Inteligência Artificial / Philipe
         Borba; orientador Edilson de Souza Bias; co-orientador
         Nilton Correia da Silva. -- Brasília, 2022.
            165 p.


            Dissertação (Mestrado - Mestrado em Geociências
         Aplicadas) -- Universidade de Brasília, 2022.


            1. Sensoriamento Remoto. 2. Deep Learning. 3. Segmentação
         Semântica. 4. Produção Cartográfica. 5. Extração da geometria
         de prédios. I. de Souza Bias, Edilson, orient. II. Correia
         da Silva, Nilton, co-orient. III. Título.
```

BANCA EXAMINADORA

Dr. Edilson de Souza Bias
Orientador

Dr. Raul Queiroz Feitosa
Membro Externo (PUC-Rio)

Dr. Weeberb João Réquia
Membro Interno (IG-UnB)

AGRADECIMENTOS

À **Cristo**, autor e consumador da minha fé, senhor e salvador. **Jesus**, aquele que deu a Sua vida para todo aquele que Nele crê possa ter relacionamento com o Pai e possa ser salvo. **Jesus Cristo**, que me sustentou, deu provisão e guiou por todos esses anos, em especial o tempo recente do mestrado. À **Maytê**, minha esposa amorosa, auxiliadora e encorajadora, que me acompanhou e me deu forças por todo esse processo. Você, Maytê, acreditou em mim, mesmo em momentos que eu mesmo duvidei. A minha familia, que me apoiou durante todo o caminho e, em especial, aos meus pais **Jorge** e **Regina** e a minha irmã **Luíza**, os quais sempre me apoiaram e investiram nos meus valores e formação acadêmica. Aos meus familiares, **Marcos**, **Márcia**, **Mayná**, **Léo**, **Mayli** e **Bruno** por todas as orações e palavras de encorajamento e conforto. Aos meus irmãos da Igreja Batista Capital, em especial ao **Pedro Magalhães** e **João Wegermann**, que me sustentaram em oração por todo esse tempo. Ao **Exército Brasileiro** que, por intermédio do **Departamento de Ciência e Tecnologia**, me confiou a missão de desenvolver uma pesquisa de interesse da Força Terrestre. À **Diretoria de Serviço Geográfico**, comandada no período dessa pesquisa pelo General de Divisão **Pedro Paulo Levi Mateus Canazio**, pelas oportunidades profissionais, confiança no trabalho e ceção dos dados utilizados na dissertação. Ao **1º Centro de Geoinformação**, que me cedeu infraestrutura de última geração para a condução da pesquisa, que me deu todo o suporte técnico e que produziu os dados utilizados. Ao **2º Centro de Geoinformação**, por todo o apoio administrativo durante essa missão. Ao meu orientador, doutor **Edilson**, que confiou, apoiou e direcionou o meu trabalho. Aos meus amigos **Diniz** e **Guimarães Filho**, agradeço pelas longas horas de discussões e retirada de dúvidas, por todo apoio técnico, pelos ensinamentos acadêmicos, e conselhos no campo profissional e pessoal: se sou o que hoje me tornei, vocês dois tem boa parcela de contribuição. Por fim, ao capitão **Pedrosa**, ao major **Dresch**, ao doutor **Matheus** (IME), ao tenente coronel **Ivanildo** (IME), ao coronel **Azeredo** (DSG), ao coronel **Barreto** (DSG), à coronel **Soraya** (MD), ao coronel **Correia**, ao tenente coronel **Emerson** (4º CGEO), e ao capitão **Fernando** (DSG) por todo o apoio acadêmico, administrativo e mentoria por todo o processo. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

RESUMO

Extração de edificações por meio de imagens oriundas de sensores aerotransportados é uma atividade essencial para aplicações como planejamento urbano e produção cartográfica. No caso da produção cartográfica, ao utilizar somente imagens, tal processo é essencialmente manual, seja por meio de restituição fotogramétrica, seja por digitalização em uma ortoimagem. Dados os recentes avanços no campo de inteligência artificial, particularmente no ramo de *Deep Learning* (DL), a presente dissertação propõe um novo método baseado em redes convolucionais profundas (*convolutional neural networks*) para extrair automaticamente a geometria de edificações em regiões densamente edificadas, denominado HRNet OCR w48 Frame Field. Tal técnica consiste na combinação entre a rede neural convolucional profunda HRNet OCR w48 e a estrutura *Frame Field*. Dessa combinação, além das máscaras binárias de segmentação, é extraído um campo vetorial complexo (Frame Field) utilizado em um método de pós-processamento denominado *Active Skeletons Method (ASM)* para a obtenção de polígonos com formatos de edificações. Além do método supramencionado, identificou-se na literatura um método que no escopo desta pesquisa foi denominado ModPolyMapper, que também é capaz de extrair polígonos de edificações e que apresentou, em regiões esparsas, resultados adequados aos parâmetros definidos na pesquisa. Em adição, a presente pesquisa também propõe um novo conjunto de dados (*dataset*), para o treinamento de métodos semelhantes aos apresentados, denominado *Brazilian Army Geographic Service Buildings dataset*. Além disso, foram desenvolvidos frameworks baseados em *software* livre e de código aberto, na linguagem de programação Python, um utilizando o Tensorflow, e outro o PyTorch, para o treinamento de redes de segmentação semântica, denominados respectivamente segmentation_models_trainer e pytorch_segmentation_models_trainer. Outrossim, também foi desenvolvido um complemento para o QGIS, denominado DeepLearningTools, para a construção e visualização de máscaras para o treinamento. Ademais, foi realizada uma análise de qualidade à luz das normas do Sistema Cartográfico Nacional (SCN) para atestar que ambos os métodos abordados nesta pesquisa são adequados para a elaboração de cartas na escala 1:25.000, em conformidade com as normas do Sistema Cartográfico Nacional (SCN).

Palavras-chave: Sensoriamento Remoto, *Deep Learning*, Segmentação Semântica, Produção Cartográfica, Extração da geometria de prédios.

ABSTRACT

Building footprint extraction using airborne imagery is essential for urban planning and cartographic production. Notably, for cartographic production, when using only images, that process is essentially handcrafted, either by photogrammetry restitution or by extracting features manually using an orthoimage. With the recent advances in Artificial Intelligence, particularly in Deep Learning, the present masters' dissertation proposes a new method based on deep convolutional neural network techniques to automatically extract building footprints in densely built-up areas, named HRNet OCR w48 Frame Field. Such a technique combines the deep convolutional neural network HRNet OCR w48 with Frame Field's structure. Besides the binary segmentation masks, these combinations extract a complex vector field (Frame Field), which is used in a post-processing technique named Active Skeletons Method (ASM) to extract building footprints. Moreover, we identified in the literature a method that in this research is named ModPolyMapper, which is also capable of extracting building footprints, and which presented relevant results according to the research parameters in sparse regions. Furthermore, the current research also proposes a new dataset for training methods similar to the previously presented ones, named Brazilian Army Geographic Service Buildings Dataset. In addition, we developed two free and open source frameworks in Python, one using Tensorflow, while the other uses PyTorch to train semantic segmentation neural networks, respectively named segmentation_models_trainer and pytorch_segmentation_models_trainer. Thus, we also developed a QGIS plugin named DeepLearningTools, used to build and visualize training masks. Additionally, we carried out a quality assurance analysis in light of the specifications of the National Cartographic System to assess that both mentioned methods in this research are suitable to build topographic charts on the scale of 1:25,000.

Keywords: Remote Sensing, Deep Learning, Semantic Segmentation, Cartographic Production, Building Footprint Extraction.

Lista de Figuras

Lista de Tabelas

Sumário

1 Introdução

Com o avanço da tecnologia, há cada vez mais satélites de imageamento modernos com alta resolução espacial, o que pode permitir um estudo mais acurado da superfície terrestre (BITTNER et al., 2018). As imagens dos referidos satélites permitem estudos de de uso e cobertura do solo (LIU et al., 2020), de detecção de mudanças da paisagem (ZHANG et al., 2019), de expansão de mancha urbana (KHANAL et al., 2019), além de ser empregáveis na produção cartográfica (HOBEL et al., 2015), dada a capacidade de se extrair informações de imagens, sendo dessa forma um dos principais insumos nas atividades de mapeamento.

Particularmente para a produção cartográfica, tal capacidade de se extrair informações é interessante pois, em tese, permite a automatização de processos e, por conseguinte, uma maior produção de Cartas Topográficas. O decreto-lei nº 243 de 28 de fevereiro de 1967 (BRASIL, 1967) define no artigo 2º o Sistema Cartográfico Nacional, que é constituído de entidades públicas e privadas, com a finalidade de realizar o mapeamento sistemático do território brasileiro, nas escalas 1:1.000.000 a 1:25.000. Dentre as entidades públicas responsáveis pelo mapeamento, pode-se citar a Diretoria de Serviço Geográfico (DSG) e o Instituto Brasileiro de Geografia e Estatística (IBGE).

O decreto-lei nº 6.666 de 27 de novembro de 2008 (BRASIL, 2008) criou a Infraestrutura Nacional de Dados Espaciais (INDE) com a finalidade de reunir dados geoespaciais de diversos órgãos governamentais e privados. Tais integrações só são possíveis por meio de normatização para a produção e disseminação de dados, as quais foram definidas no âmbito da INDE, em concordância com a legislação do SCN supramencionada.

A Comissão Nacional de Cartografia (CONCAR) era o órgão responsável por coordenar a elaboração de normas de produção e disseminação de dados geoespaciais no âmbito da INDE. Atualmente, a CONCAR foi extinta pelo governo atual e encontra-se em processo de reestruturação.

A DSG e o IBGE, dentre outros órgãos, faziam parte da CONCAR e integram a INDE. Apesar da descontinuidade da CONCAR, as normas para produção e disseminação de dados geoespaciais ainda encontram-se em vigor. Logo, para que dados produzidos se tornem dados de referência no SCN, é obrigatório que as informações obtidas sejam compatíveis com as normas e padrões nacionais definidos pela CONCAR abaixo relacionadas:

- Especificação Técnica para Produtos de Conjuntos de Dados Geoespaciais (ET-PCDG) (DSG, 2016b): especifica as características técnicas dos produtos do SCN, bem como os metadados;

- Especificação Técnica para Estruturação de Dados Geoespaciais Vetoriais (ET-EDGV) (CONCAR, 2010): define o modelo de dados oficial brasileiro;

- Especificação Técnica para Aquisição de Dados Geoespaciais Vetoriais (ET-ADGV) (DSG, 2018): define as regras de aquisição da geometria dos dados, bem como os tamanhos mínimos dos objetos; e

- Especificação Técnica para Controle de Qualidade dos Produtos de Conjuntos de Dados Geoespaciais (ET-CQDG) (DSG, 2016a): define os procedimentos para aferir a qualidade dos dados produzidos.

Em função de vários fatores como por exemplo a extensão territorial continental do Brasil, no SCN há uma situação de desatualização ou até mesmo falta de Cartas Topográficas em diversas escalas como 1:50.000 e 1:25.000, constituindo-se um grande desafio para o mapeamento nacional.

Especificamente na escala 1:25.000, dos 46.391 produtos dessa escala previstos para o recobrimento do território brasileiro, há apenas 1.447 produtos disponíveis no Banco de Dados Geográficos do Exército (BDGEx) (DSG, 2010), representando 3,12% do total. Tal percentagem pode ser explicada pela dificuldade de mapeamento da escala 1:25.000, uma vez que de todas as escalas previstas para mapeamento no SCN (1:1.000.000 a 1:25.000), conforme descritos no decreto-lei nº243 (BRASIL, 1967) e especificado pela ET-PCDG (DSG, 2016b), a escala em questão é a que contém maior detalhamento, demandando para sua confecção imagens de maior resolução espacial. Com isso, os produtos que requerem maior investimento tanto financeiro para a aquisição de insumos, quanto de recursos humanos para a elaboração.

Atualmente, no âmbito dos projetos de mapeamento da DSG, parte das informações necessárias para a elaboração de produtos cartográficos é extraída manualmente, como as geometrias de edificações e de áreas edificadas. Nesse contexto, a pesquisa e desenvolvimento de métodos automáticos de extração de informação de imagens de satélite na produção cartográfica pode vir a acelerar algumas etapas do processo produtivo da DSG.

Para a extração de informações de imagens oriundas de sensoriamento remoto, tradicionalmente, utiliza-se de técnicas de segmentação baseadas em intervalos de histograma, em agrupamento de pixels, em regiões e em identificação de bordas (BLASCHKE, 2010; CHEN et al., 2018). Além disso, podem-se elencar sem exaurir todas as técnicas existentes, segmentação e classificação baseada em objeto, tradução do termo *object based image analysis* (OBIA), análise de textura, de reflectância e de formato (SU et al., 2008; WISEMAN et al., 2009; MA et al., 2017; KUCHARCZYK et al., 2020).

Como evolução dos conhecimentos supramencionados, pode-se encontrar na literatura, como em Li et al. (2014), em Damodaran et al. (2017), Thanh Noi e Kappas (2017), em Jozdani et al. (2019) e em Li et al. (2020), aquelas conhecidas como técnicas de aprendizado de máquina utilizadas no Sensoriamento Remoto, como *Support Vector Machines* (SVM), *Random Forests* (RF) e *Decision Trees* (DT).

Em adição, nos últimos anos também ocorreram avanços tecnológicos no campo da Computação, particularmente no campo da Inteligência Artificial (IA). Tais evoluções podem ser explicadas, em parte, pela melhoria do poder computacional e pelo advento de novos algoritmos de processamento de dados utilizando placas de vídeo (*GPU*) (CHEN, 2016; BALL et al., 2017). Nesse contexto, uma técnica que vem sendo largamente empregada em diversos estudos científicos é denominada *Deep Learning* (DL).

Com o advento dos algoritmos de DL, a comunidade científica passou a se interessar por esse tipo de conhecimento dada a maior acurácia resultante, como pode ser verificado nos estudos Adarme et al. (2020), Mboga et al. (2019), Guirado et al. (2017), nos quais técnicas clássicas das geociências utilizadas para extrair feições por meio de imagens de sensoriamento remoto foram comparadas com aplicações utilizando o Estado da Arte de DL.

Segundo Lecun et al. (2015), Hoeser e Kuenzer (2020), ao contrário dos métodos que são baseados apenas nos valores de pixels para identificar padrões, o aspecto chave de DL é que este método se vale de múltiplas camadas de abstração obtidas por processos matemáticos não lineares que permitem a inferência de informações escondidas, conjunto de técnicas de IA conhecidas como redes convolucionais profundas (*deep convolutional neural networks*). Pode-se citar como características desse tipo de algoritmo, a necessidade de muitos dados para o treinamento e a exigência de alto poder computacional (NAJAFABADI et al., 2015).

No contexto das geociências, há diversos problemas que possuem soluções propostas utilizando *Deep Learning*, como detecção de objetos (MUSYAROFAH et al., 2020; LI et al., 2020; CHENG et al., 2016; HU et al., 2015; CHEN et al., 2019a), classificação de imagens hiperespectrais (ZHU et al., 2017; DATA et al., 2015; AUDEBERT et al., 2019; PAN et al., 2020), estudo de uso e cobertura do solo (ALHASSAN et al., 2020; JOZDANI et al., 2019; CARRANZA-GARCÍA et al., 2019), superresolução (LANARAS et al., 2018; QIN et al., 2020; SALVETTI et al., 2020), detecção de mudanças (SONG; CHOI, 2020; KULKARNI; VENUGOPAL, 2020; LIU et al., 2019; WANG et al., 2018) e, finalmente, segmentação semântica (DIAKOGIANNIS et al., 2019; ZHANG et al., 2016; KEMKER et al., 2018; YANG et al., 2018), que é uma técnica de visão computacional que permite ao computador reconhecer e extrair objetos de imagens

(SHAPIRO, 2001), e se adequa ao caso de uso de extração automática de feições para a elaboração de produtos cartográficos.

Diante dos avanços elencados e dadas as necessidades de mapeamento supramencionadas, uma pesquisa cujo resultado fosse aplicar o Estado da Arte de DL em atividades de mapeamento seria justificável. Porém, para que informações resultantes do uso de técnicas de DL possam ser utilizadas na produção cartográfica, algumas operações de generalização cartográfica são necessárias para pós-processar os dados inferidos devido à geração de feições com imperfeições como serrilhamento, excesso de vértices e e com formato geométrico que nem sempre são compatíveis com as regras de aquisição definidas pelo SCN. Trabalhos como Partovi et al. (2017), Li et al. (2017), Lokhat e Touya (2016) tratam de algumas formas de generalizar os dados obtidos e demonstram a necessidade de pesquisas sobre esses procedimentos.

Sendo assim, identifica-se um problema científico de encontrar uma metodologia em que se possa utilizar arquitetura de rede neural convolucional profunda, treinada com um conjunto de dados adequado ao problema, para extrair informações geográficas no formato vetorial que possam representar as formas poligonais de feições no terreno como por exemplo, corpos dágua, vegetação, terreno exposto, áreas densamente edificadas e edificações.

Além disso, para que tais informações possam ser utilizadas em atividades de mapeamento, as geometrias resultantes devem ter acurácia compatível com normas do SCN para a escala 1:25.000. Logo, uma pesquisa cujo objetivo seja investigar formas de utilizar inteligência artificial para realizar tais extrações para a cartografia se justifica.

Portanto, considerando-se apenas o problema de extração automática de edificações, a presente dissertação de mestrado tem como proposta de hipótese científica que **é possível treinar uma rede convolucional profunda que seja capaz de segmentar e extrair feições vetoriais no formato poligonal de edificações, geometricamente consistentes, com acurácia compatível com a escala 1:25.000, em conformidade com as normas definidas pelo SCN**.

2 Objetivo

Essa pesquisa tem por objetivo verificar se é possível treinar uma rede neural convolucional profunda, que seja capaz de segmentar imagens de satélite de altíssima resolução, com a finalidade de extrair geometrias de edificação com características e níveis de qualidade aderentes aos padrões definidos pelo SCN, na escala 1:25.000.

2.1 Objetivos Específicos

Como objetivos específicos da pesquisa, pretende-se:

(a) Identificar arquiteturas de redes neurais capazes de extrair informações para a obtenção de geometrias de edificação;

(b) Verificar qual é a melhor arquitetura para ser utilizada em atividade de mapeamento, levando-se em consideração o tempo de processamento em relação a acurácia obtida;

(c) Identificar o melhor método, ou combinação de métodos, para gerar os polígonos a partir das redes neurais treinadas;

(d) Desenvolver uma metodologia de treinamento de rede neural convolucional profunda, preferencialmente materializada por meio de automações desenvolvidas no âmbito da pesquisa;

(e) Desenvolver soluções baseadas em software livre para automatizar a metodologia de treinamento proposta na pesquisa; e

(f) Avaliar se os polígonos obtidos ao final do processo de extração possuem acurácia compatível para a elaboração de produtos cartográficos na escala 1:25.000.

3 Organização da Pesquisa

Para atingir os objetivos definidos na seção 2, foi conduzida uma pesquisa dividida em cinco partes, as quais podem ser visualizadas com detalhe na figura 1:

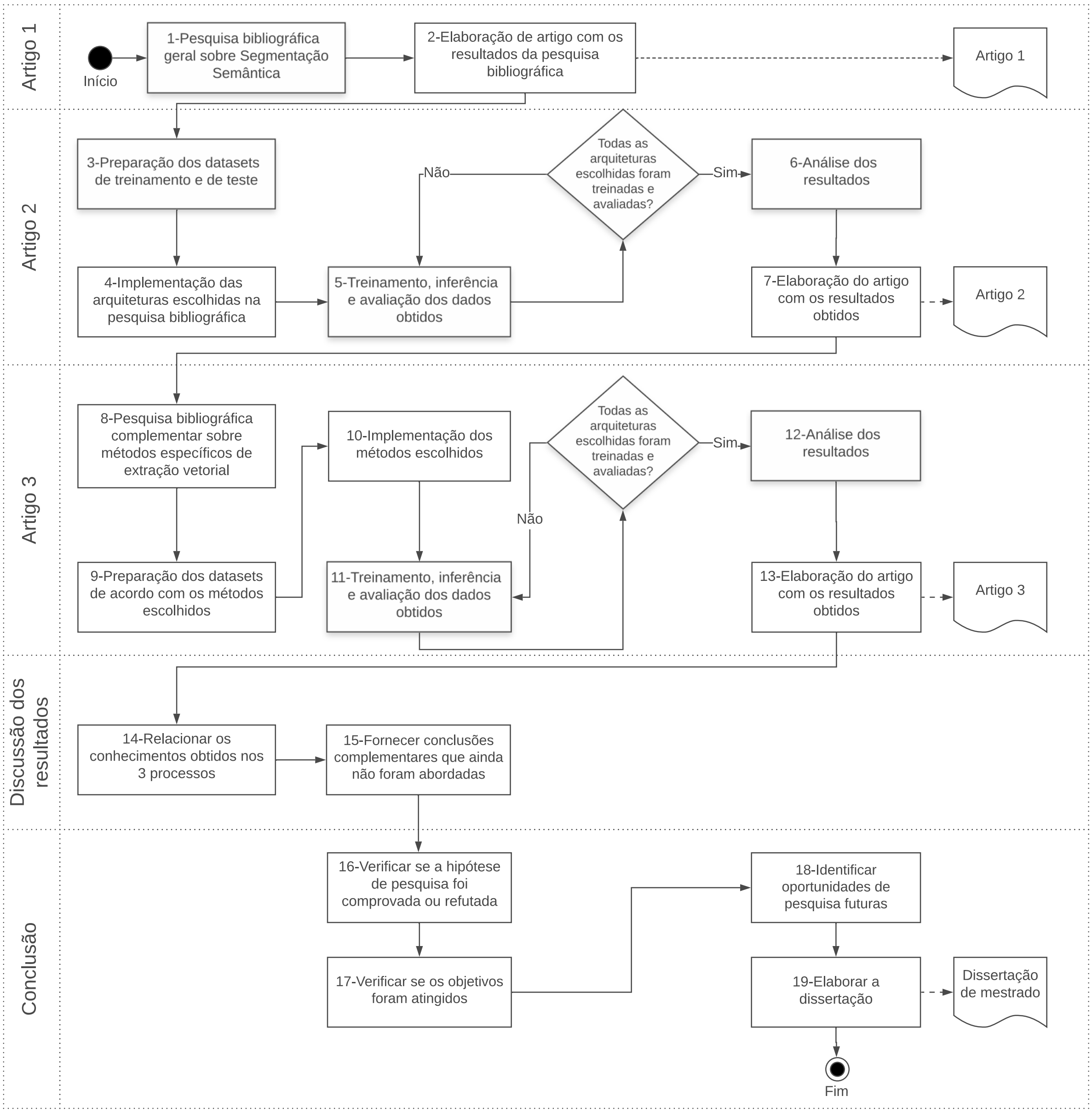

Artigo 1
Início
1-Pesquisa bibliográfica geral sobre Segmentação Semântica
2-Elaboração de artigo com os resultados da pesquisa bibliográfica
Artigo 1
Artigo 2
3-Preparação dos datasets de treinamento e de teste
Não
Todas as arquiteturas escolhidas foram treinadas e avaliadas?
Sim
6-Análise dos resultados
4-Implementação das arquiteturas escolhidas na pesquisa bibliográfica
5-Treinamento, inferência e avaliação dos dados obtidos
7-Elaboração do artigo com os resultados obtidos
Artigo 2
Artigo 3
8-Pesquisa bibliográfica complementar sobre métodos específicos de extração vetorial
10-Implementação dos métodos escolhidos
Todas as arquiteturas escolhidas foram treinadas e avaliadas?
Sim
12-Análise dos resultados
Não
9-Preparação dos datasets de acordo com os métodos escolhidos
11-Treinamento, inferência e avaliação dos dados obtidos
13-Elaboração do artigo com os resultados obtidos
Artigo 3
Discussão dos resultados
14-Relacionar os conhecimentos obtidos nos 3 processos
15-Fornecer conclusões complementares que ainda não foram abordadas
Conclusão
16-Verificar se a hipótese de pesquisa foi comprovada ou refutada
18-Identificar oportunidades de pesquisa futuras
17-Verificar se os objetivos foram atingidos
19-Elaborar a dissertação
Dissertação de mestrado
Fim

Figura 1: Fluxograma da pesquisa.

1. A primeira parte refere-se ao levantamento do Estado da Arte de Segmentação Semântica, de forma que se possa reconhecer quais métodos podem resolver o problema;

2. A segunda parte consiste em realizar uma aplicação de algumas das técnicas estudadas na parte 1, para que se possa identificar se tais métodos são adequados

para a extração de edificações. O critério para a seleção das técnicas estudadas será descrito na seção que abordará esse tópico;

3. A terceira parte consiste em estudar métodos complementares para extração dos polígonos de edificações, implementar cada um deles, realizar os treinamentos necessários e avaliar os resultados;

4. A quarta parte consiste em relacionar as ideias apresentadas nos três primeiros artigos. Além disso, pretende-se complementar pontos que não puderam ser aprofundados nos artigos. Cada informação complementar será disposta em uma subseção posterior à apresentação de cada artigo, com a finalidade de encadear as ideias apresentadas para o leitor;

5. Por fim, a quinta parte tem por finalidade identificar se a hipótese de pesquisa e se os objetivos foram atingidos. Além disso, nesta seção pretende-se fornecer conclusões complementares à pesquisa e sugerir pesquisas futuras.

Diante do exposto, a presente dissertação tem a seguinte estrutura:

(a) Capítulo 4: levantamento e análise do estado da arte de Segmentação Semântica em aplicações de sensoriamento remoto, realizado por meio do artigo de título ”*A Review of Remote Sensing Applications on Very High-Resolution Imagery Using Deep Learning-Based Semantic Segmentation Techniques*”, publicado no International Journal of Advanced Engineering Research and Science (BORBA et al., 2021a) e apresentado na seção 4.1. Além dsso, discussões e informações complementares sobre tal publicação serão realizadas na seção 4.2;

(b) Capítulo 5: Aplicações das técnicas levantadas no capítulo 4 apresentadas no artigo de título ”*Building Footprint Extraction using Deep Learning Semantic Segmentation Techniques: Experiments and Results*”, publicado nos anais do *International Geoscience and Remote Sensing Symposium 2021* (IGARSS 2021) (BORBA et al., 2021b). Ademais, as discussões complementares e os próximos passos da pesquisa serão realizados, na subseção 5.1;

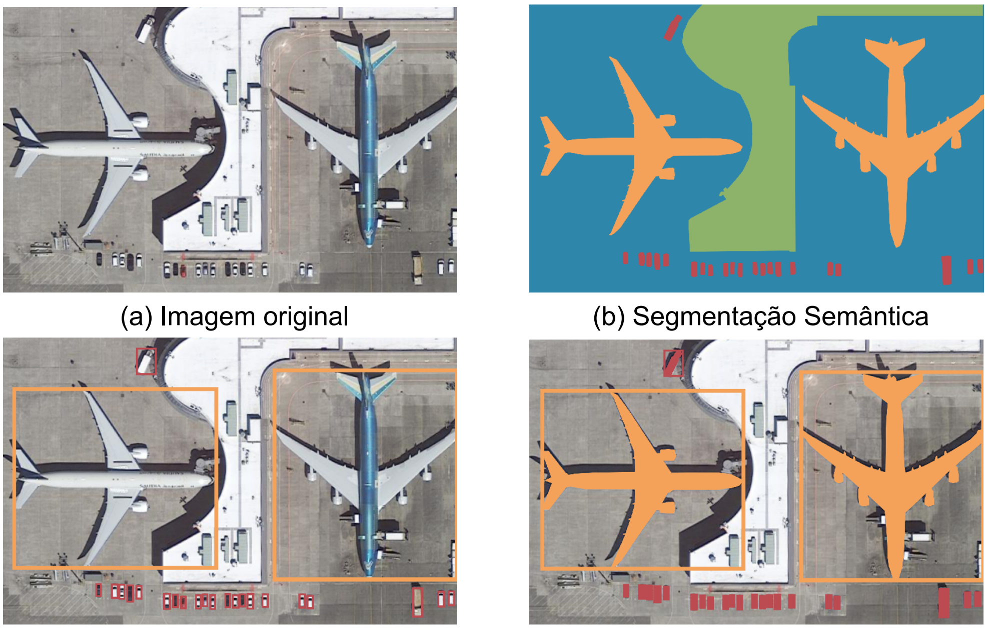
(c) Capítulo 6: Resultados da pesquisa, apresentados na subseção 6.1 por meio do artigo de título ”*Building Polygon Extraction from Very High-Resolution Remote Sensing Images using Deep Learning Methods: A meta-analysis with an experimental approach*”, a ser submetido ao *ISPRS Journal of Photogrammetry and Remote Sensing*. Na subseção 6.2, serão realizadas discussões complementares e considerações relativas aos padrões de qualidade relativos ao SCN;

(d) Capítulo 7: Conclusão da pesquisa. Nesse último capítulo, será verificado se a hipótese de pesquisa foi comprovada ou refutada e se os objetivos foram alcançados. Além disso, são apontadas oportunidades de pesquisas futuras.

4 Estado da Arte de Segmentação Semântica

Conforme foi ilustrado na introdução desta dissertação, há diversos avanços nas pesquisas que utilizam *Deep Learning* em aplicações de Sensoriamento Remoto (SR). Particularmente para extração de geometrias por meio de imagens de SR, segundo Hoeser e Kuenzer (2020) pode-se utilizar técnicas de Segmentação Semântica (SS) e Segmentação de Instância (SI).

Métodos que utilizam SI são subconjuntos de SS, dado que para realizar SI deve-se fazer a localização e posterior SS da região identificada. Um resumo de técnicas apontadas por Hoeser e Kuenzer (2020) pode ser visualizado na figura 2.



(a) Imagem original

(b) Segmentação Semântica

(c) Detecção de Objetos

(d) Segmentação de Instância

Figura 2: Problemas de extração de informações em imagens de satélite. Adaptado de Hoeser e Kuenzer (2020).

Diante dos problemas ilustrados na figura 2 e diante da escala de trabalho (1:25.000) definida na hipótese científica, decidiu-se estudar técnicas apenas de SS, visto que de acordo com as regras definidas na ET-ADGV (DSG, 2018), edificações vizinhas são aglutinadas em apenas um elemento. Além disso, dependendo da técnica escolhida, as de SS tendem a ser menos computacionalmente custosas, em comparação às de SI.

Os métodos de Segmentação Semântica considerados na presente pesquisa são redes neurais convolucionais e, para que se possa utilizar tais métodos, deve-se realizar um processo chamado treinamento, o qual segundo Goodfellow et al. (2016), consiste em um processo iterativo, em que é dado uma entrada para a rede e é calculada a saída por meio de operações de convoluções combinadas a operações de redução de dimensionalidade denominadas *pooling* e de inserção de não-linearidades no processo por meio do uso de funções de ativação. A diferença entre o resultado e a saída esperada é medida por meio de uma função, denominada função de perda (*loss function*). Em seguida, os pesos da rede (valores numéricos de todos os *kernels* das convoluções) são atualizados de acordo com a *loss* por meio de um processo chamado *back propagation*. O processo brevemente explicado pode ser visualizado na figura 3.

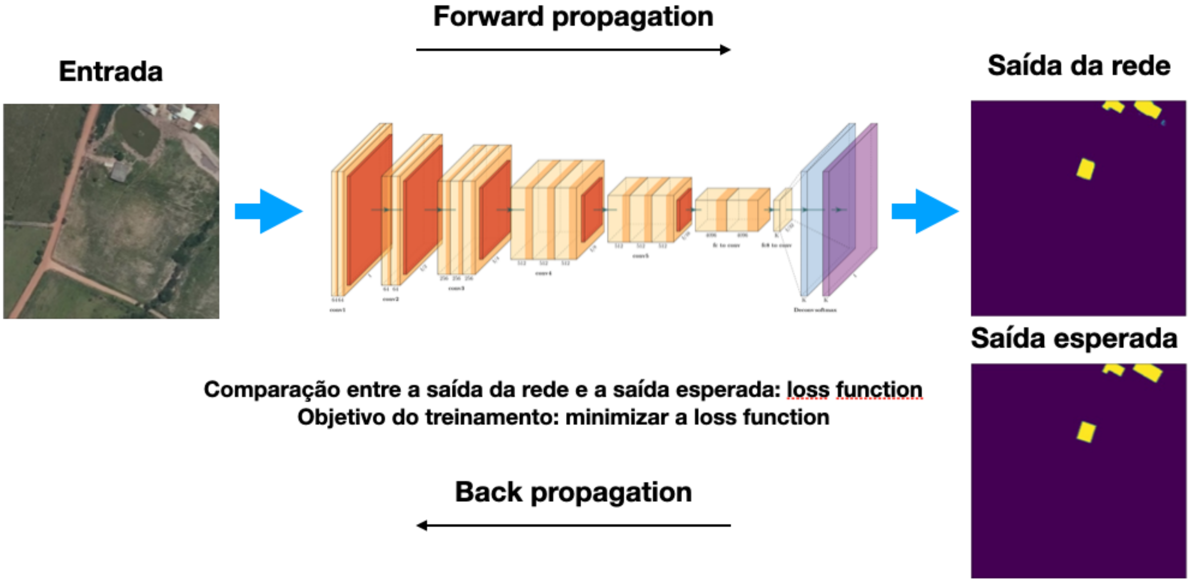

Forward propagation
Entrada
Saída da rede
Saída esperada
Comparação entre a saída da rede e a saída esperada: loss function
Objetivo do treinamento: minimizar a loss function
Back propagation

Figura 3: Esquema de treinamento ilustrando os processos de *forward propagation* e *back propagation*. Ao longo do *forward propagation*, normalmente as arquiteturas de redes neurais convolucionais profundas empilham diversos resultados de operações de convolução e aplicam redução de dimensionalidade por meio de operações de *pooling*, além de inserir não linearidades utilizando as funções de ativação. A representação esquemática da rede foi gerada utilizando os códigos disponíveis em https://github.com/HarisIqbal88/PlotNeuralNet.

Tendo em vista que nesse momento da pesquisa, fazia-se necessário entender os pormenores dos processos de treinamento, as estruturas envolvidas e os dados necessários para a realização do treinamento, foi necessário realizar um processo de levantamento de técnicas existentes.

Além disso, era essencial identificar as diferentes formas de realizar as operações de convolução para obter as máscaras segmentadas. Às maneiras de se combinar as operações supramencionadas se dá o nome de arquitetura. Uma parte especial das arquiteturas de redes neurais convolucionais são os *backbones*, os quais são responsáveis pela extração de padrões. Existe uma grande quantidade de *backbones* propostos na literatura, sendo assim, também era interessante estudar quais existiam, bem como identificar quais são os mais populares.

Sendo assim, foi realizada uma pesquisa do estado da arte de aplicações de Segmentação Semântica utilizando técnicas de *Deep Learning* em Sensoriamento Remoto, a qual visava atingir os seguintes objetivos:

(a) Identificar as principais arquiteturas e *backbones* utilizados, apontando, se possível, quais são os mais famosos;

(b) Identificar as melhores combinações de arquiteturas e *backbones* em diferentes conjuntos de dados (*datasets*);

(c) Descrever o processo de treinamento, apontando cada uma das componentes básicas das arquiteturas (como convoluções, *pooling* e funções de ativação), bem como algumas funções de perda (*loss functions*) normalmente utilizadas;

(d) Identificar métricas de avaliação dos resultados dos treinamentos;

(e) Identificar técnicas de melhorar os resultados dos treinamentos;

(f) Identificar alguns conjuntos de dados para treinamento;

(g) Identificar as bibliotecas disponíveis para a implementação das arquiteturas e realização dos treinamentos; e

(h) Identificar em aplicações de Ciência da Computação oportunidades de pesquisa em Sensoriamento Remoto.

O presente capítulo é organizado da seguinte forma:

- A subseção 4.1 apresenta o artigo de título ”*A Review of Remote Sensing Applications on Very High-Resolution Imagery Using Deep Learning-Based Semantic Segmentation Techniques*”, publicado no International Journal of Advanced Engineering Research and Science (IJAERS) (BORBA et al., 2021a), no qual foi realizado o levantamento do estado da arte de aplicações de técnicas de segmentação semântica em publicações de sensoriamento remoto; e

- A subseção 4.2, na qual é fornecida uma discussão complementar dos resultados obtidos.

4.1 Artigo de Revisão

International Journal of Advanced Engineering Research and Science (IJAERS)
Peer-Reviewed Journal
ISSN: 2349-6495(P) | 2456-1908(O)
Vol-8, Issue-8; Aug, 2021
Journal Home Page Available: https://ijaers.com/
Article DOI: https://dx.doi.org/10.22161/ijaers.88.29

A Review of Remote Sensing Applications on Very High-Resolution Imagery Using Deep Learning-Based Semantic Segmentation Techniques

Philipe Borba[1,2], Edilson de Souza Bias[2], Nilton Correia da Silva[3], Henrique Llacer Roig[2]

[1]Brazilian Army Geographic Service, Brazil
[2]Geosciences Institute, University of Brasília, Brazil
[3]Campus Gama, University of Brasília, Brazil

Received: 07 Jul 2021,

Received in revised form: 05 Aug 2021,

Accepted: 12 Aug 2021,

Available online: 22 Aug 2021

***Keywords*—Remote Sensing, Deep Learning, Semantic Segmentation, Convolutional Neural Networks, State-of-the-art, Review.**

***Abstract*—*Semantic Segmentation is a technique in Computer Sciences (CS) to extract information from images. Recent advances in Artificial Intelligence, particularly in Deep Learning, Semantic Segmentation combined with techniques such as convolutional neural networks, have presented better results and exciting results. Due to its power and better results than classical approaches, there has been an increase in research articles in Remote Sensing that propose using deep learning-based semantic Segmentation to extract information from satellite or airborne imagery. In this paper, we surveyed the state-of-the-art of Semantic Segmentation in Remote Sensing from 2010 until 2020 by identifying the research topics and the number of publications and citations. Furthermore, we also pointed out the fundamental algorithms, the main convolutional neural network architectures, backbones, and the most used evaluation metrics. In addition, some datasets were highlighted, as well as some frameworks that can be used to train semantic segmentation deep neural networks. Finally, we have shown some applications of the showcased techniques and concluded the paper by pointing out some research opportunities of Remote Sensing Semantic Segmentation, concerning some bleeding-edge scientific papers published in 2020 in CS.*

I. INTRODUCTION

The extraction of information from remote sensing images has been an active research field, with essential applications for urban planning, urban dynamics modeling, and disaster damage assessment. Semantic Segmentation is the process of assigning a label to each pixel of an image and decompose a scene into semantically meaningful regions [1]. Traditionally, semantic Segmentation is performed either pixel-wise or with object-based approaches. The latter is known as Geographic Object-Based Image Analysis (GEOBIA) [2] and usually outperforms the former. These approaches typically consist of two separate steps: Segmentation followed by classification. Because the second step's accuracy usually relies on the first step's quality, image segmentation is critical for GEOBIA.

However, image segmentation is not a trivial task, given that most algorithms rely on subjective and arbitrary parameters setting. The incorrect choice of parameters may lead to undesired results, such as under-segmentation and over-segmentation, which may impact the classification accuracy. Moreover, segmentation techniques' generalization capability is limited because they cannot deal with the objects' complexity present in an image. For example, a given set of parameters can provide good segmentation results at homogeneous regions (e.g., agricultural fields) and unsatisfactory results in heterogeneous areas like urban environments.

Thus, image analysts usually try several parameter combinations to achieve a suitable outcome for an entire scene, a time-consuming task. Adaptive segmentation algorithms were proposed to deal with the diversity of image objects [3, 4] or automatic tuning of segmentation parameters [5, 6]. However, these methods are complex, rely on human-made reference images, and are designed for specific applications.

Recently, improvements in computation power and parallel processing algorithms using graphics processing units (GPUs) favored the development of deep learning (DL) [7, 8], particularly convolutional neural networks (CNNs), a type of DL method introduced by [9], have become exceedingly popular for classification, object localization, and semantic segmentation of remote sensing images [10]. CNNs are designed to automatically extract spatial patterns (e.g., shapes, edges, texture) of images using a set of convolutions and pooling operations, hence learning object-specific characteristics in an end-to-end fashion.

Particularly in the context of semantic Segmentation, neural networks have achieved outstanding results [11, 12, 13, 14, 15, 16, 17, 18]. Unlike traditional pixel-wise classification, semantic Segmentation using CNNs can preserve the object boundaries producing sharp, fine-scale Segmentation. Fully convolutional networks (FCNs) were the first approach that employed deep networks for semantic Segmentation. The rationale behind FCNs relies on transforming the fully connected layers into upsampling or transposed convolutional layers [19] to perform dense pixel predictions. The pioneering work of [19] adapted well-known CNNs models such as AlexNet for semantic segmentation tasks.

In semantic Segmentation, the smallest segment can be a single pixel, which is not adequate for most applications of information extraction using high-resolution remote sensing images because, in these images, it is improbable to find a target with the dimensions of a single pixel. To overcome this problem, instance segmentation combined object detection and semantic segmentation can be used to classify an object at the pixel level and outline its exact shape [20]. Both semantic Segmentation and instance segmentation networks provide the opportunity to simultaneously detect and classify building footprints without the need for a previous segmentation step, thus vanquishing the limitations of GEOBIA.

This paper will cover the latest state-of-the-art (SOTA) of semantic Segmentation in very high-resolution remote sensing, focusing only on methods that use convolutional neural networks (CNNs). We also want to identify research opportunities in RS by briefly analyzing the latest trends on CS. To fulfill this goal, this review is organized as follows: in section 2, we show the SOTA of semantic Segmentation in RS and CS papers; in section 3, we cover the basic concepts of DL and semantic segmentation techniques, the primary neural network architectures, the available datasets and frameworks and finally some raster to vector methods; and in section 4 we sum up the concepts presented in this paper, as well as cover the opportunities of research in geosciences based on the comparison of the SOTA semantic segmentation methods.

II. LITERATURE REVIEW

We conducted a literature review on remote sensing to identify the most relevant deep learning techniques and methods employed to extract information from remote sensing imagery, presented in section 2.1.

Moreover, to identify possible new techniques from computer sciences, we carried out a brief literature survey on review articles and also pointed out the best results on popular benchmarks showcased on Papers With Code [21], shown in section 2.2.

2.1. Literature Review on Remote Sensing

To perform our literature review, we searched the knowledge database SciELO Citation Index (Web of Science) to investigate further what are the main research topics, the number of publications per year, and the most cited papers. This information was used to try to delineate the most relevant papers so that we could further analyze them so that we could extract more helpful information, such as the most popular methods employed.

The term” Semantic Segmentation” was searched using the time range 2010-2020 as the filter, and there were 10,145 results, then were filtered once more, considering only the” Remote Sensing” field, yielding 718 results. To identify the main research topics, we built a word cloud, shown in figure 1, with the keywords of these results. Analyzing the picture, we can infer that the research conducted from 2010 until 2020 has used neural networks, particularly convolutional neural networks (CNNs), to extract or identify features using high-resolution satellite or aerial imagery. Common ground features extracted by the considered papers are roads and buildings.

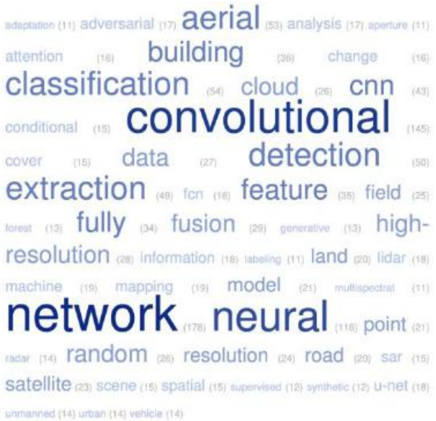


Fig. 1: Word cloud built with the keywords of the results of the search Semantic Segmentation on the Web of Science database, from 2010 to 2020, considering only papers in Remote Sensing. Larger words mean more recurring terms in the research papers' keywords.

During the considered time range, there has been a nearly exponential growth in the number of papers in remote sensing that covers semantic Segmentation that can be visualized in figure 2. The years 2015 and 2016 have presented a slight increase in the number of publications that might be a consequence of the papers published in CS, such as [22, 23, 24]. From 2017 until 2019, there has been a significant increase in the number of research papers, peaking at 140 in 2019. Since 2020 is not over yet, we can expect an even more substantial number than 2019, since the number of research papers published in 2020 is much higher than 2018's and only 40% smaller than2019's.

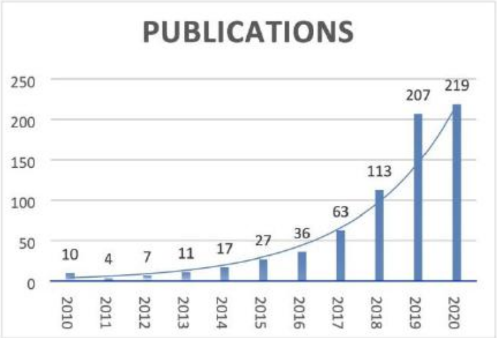


Fig. 2: Number of publications in Remote Sensing with the subject Semantic Segmentation from 2010 to 2020 registered on Web of Science.

We further narrowed our chosen papers by cross-referencing our search results with data from a GitHub repository (https://github.com/thho/DLinEO_review), which is under the license CC-BY-4.0 and contains data used in [1, 25]. Using this info, we have only considered semantic Segmentation, resulting in 261 papers to analyze. Then, we built the graph in figure 3 to find out the most popular architecture. We concluded that the most famous architecture in RS papers is the U-Net, followed by custom architectures and then Fully Convolutional Networks (FCNs).

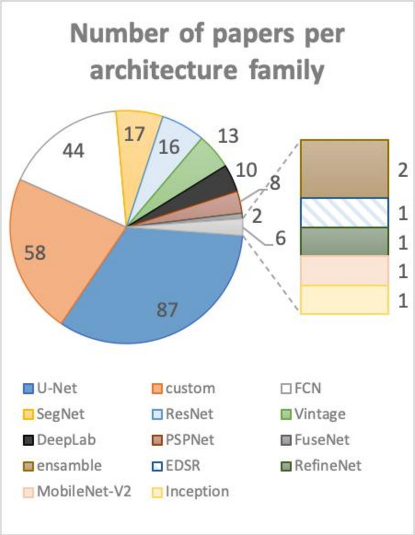


Fig. 3: Papers grouped by architecture family.

Then, to evaluate the backbone usage, we built a word cloud shown in figure 4 to find out the most popular backbones, and we found out that ResNets, VGG-16, and the Inception series are very popular.

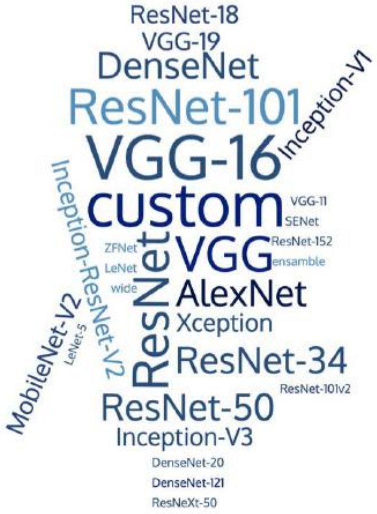


Fig. 4: Family architectures used in Semantic Segmentation papers in Remote Sensing in the considered papers. Larger names represent more popular family architecture.

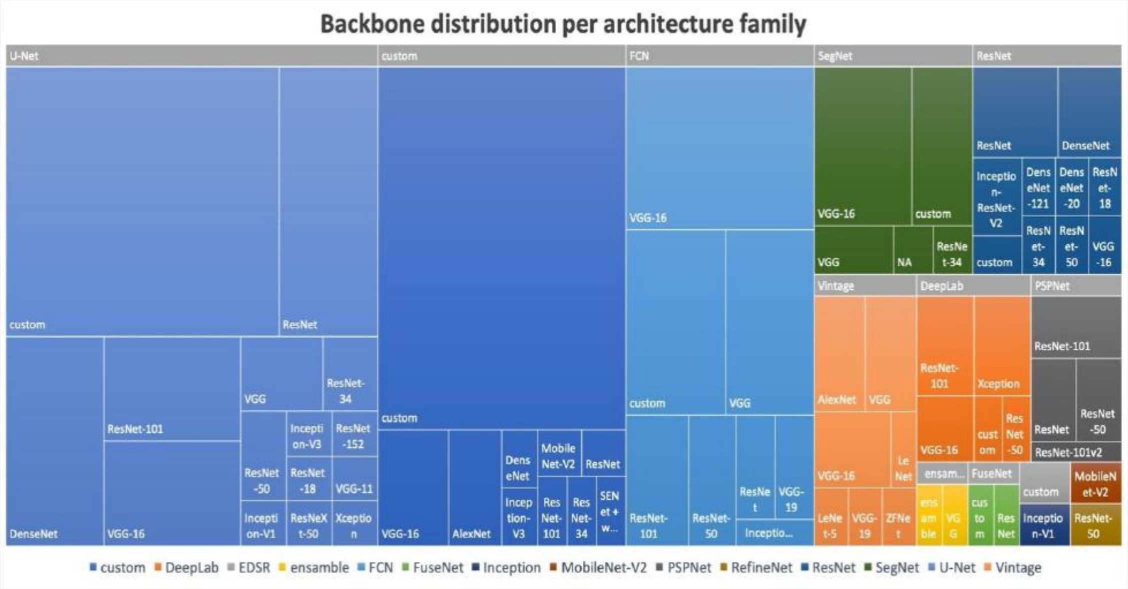


Fig 5: Tree Map representing the backbone distribution for each type of convolutional neural network architecture used in the considered papers.

To understand the relationship between the backbones and the architectures chosen in each paper and presented in the data here analyzed, we built a tree map shown in figure 5, which leads us to conclude that U-Nets with custom and ResNet backbones are very popular, followed by custom backbone and custom architecture, then by VGG-16 backbone with FCN architecture, and finally, VGG-16 backbone with SegNet architecture.

2.2. Brief Literature Review on Computer Science

There are several review articles in Computer Sciences [26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42] that portray the evolution of deep learning-based semantic segmentation methods. Common research fields on CS that use the mentioned techniques are research on self-driving vehicles [43, 44], pedestrian detection [45, 46] and computer aided diagnosis using medical images [47, 48].

The surveyed papers cover similar architectures and backbones already listed on 2.1. The novel backbones that were not identified in section 2.1 are the ones from the EfficientNet family, ResNeSt [49], and SE-ResNet family [50]. The training datasets used in CS applications are one of the main differences from RS studies. As examples of common datasets used in CS, we can cite the Cityscapes dataset [51], the PASCAL VOC (PASCAL Visual Object Classes Challenge) [52], and its extension, the PASCAL Context [39].

There is a platform called Papers With Code [21] that gathers results of several papers, as well as codes that are available online to reproduce such study considered papers. On this website, the results of each benchmark are ranked, and the best models are presented. Some of the models with the best results on the previously mentioned datasets are shown in table 1:

Table 1: Best models on some available datasets, according to Papers With Code [21].

Dataset	Best Model	Paper Title	mIoU
Cityscapes test	HRNet-OCR	Hierarchical MultiScale Attention for Semantic Segmentation [53]	85.1%
PASCAL VOC 2012 test	EfficientNet-L2+NAS-FPN	Rethinking Pretraining and Self-training [54]	90.5%
PASCAL Context	Channelized Axial Attention (CAA) with Simple decoder (Efficientnet-B7)	Channelized Axial Attention for Semantic Segmentation [55]	60.5%
Cityscapes val	HRNetV2-OCR+PSA	Polarized SelfAttention: Towards High-quality Pixelwise Regression [56]	86.95%

Other worth mentioning techniques found on the cited review papers and the research shown in table 1 are self-training [57], Channelized Axial Attention [55], and

Polarized Self-Attention [56].

III. MAIN CONCEPTS AND METHODOLOGIES IN SEMANTIC SEGMENTATION

From the SOTA review carried out in section 2, we identified some of the main concepts and techniques that we need to understand when studying semantic segmentation techniques applied to remote sensing.

Furthermore, considering the selected papers and regarding the ideas highlighted in the SOTA review, we will present some basic concepts in section 3.1, some training improving techniques in section 3.2, the main convolutional neural network backbones in section 3.3, the main architectures on section 3.4, some applications on RS and examples of some available datasets on section 3.5, and finally, some frameworks and tools on section 3.6.

3.1. Main Concepts of Convolutional Neural Networks

The convolution layer is one of the building blocks of Deep Learning. It can be defined as a combination of linear and nonlinear operations such as convolution and activation functions [58].

Convolution is a mathematical operation that applies an array of numbers (kernel) to the input, enabling feature extraction operations [58]. On the other hand, the activation function is a mathematical resource to introduce nonlinearities in the convolutional neural networks. Some examples of them are the sigmoid function, the hyperbolic tangent function, the rectified linear unit (ReLU) [58], the leaky rectified linear unit (Leaky ReLU) [59], the exponential linear unit (ELU) [60], the scaled exponential linear unit (SELU) [61], the gaussian error linear unit (GELU) [62], the Mish [63] and the Softmax [64]. Their mathematical definitions can be seen, respectively, on equations 1, 2, 3, 4, 5, 6, 7, 8, and 9. It is worth mentioning that Softmax is often used as an output function on convolutional neural networks.

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

$$tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \tag{2}$$

$$ReLU(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{3}$$

$$Leaky\_ReLU(x) = \begin{cases} 0.01x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{4}$$

$$ELU(x) = \begin{cases} \alpha(e^{x}-1) & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases} \tag{5}$$

$$SELU(x) = 1.597 \begin{cases} 1.67326(e^{x}-1) & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{6}$$

$$GELU(x) = 0.5x\left(1 + tanh\left(\sqrt{\frac{2}{\pi}}\left(x + 0.044715x^{3}\right)\right)\right) \tag{7}$$

$$Mish(x) = x \cdot ln(1+e^{x}) \tag{8}$$

$$\text{Softmax}(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \tag{9}$$

The difference between filters that use convolutions (common in image processing tasks) and the convolutional layers of CNNs is that, instead of applying a pre-determined kernel to the input, it learns the best parameters of the kernel to extract features due to the training process [33, 39, 34].

Another critical concept in CNN theory is the pooling layer, which replaces a small neighborhood of a feature map with some statistical information, such as mean or max [39]. This process is vital because it sub-samples images, reducing the dimensionality of the feature maps by introducing a translation invariance to small shifts and distortions and decreasing the number of learnable parameters [58].

The combination of convolutional layers, activation functions, and pooling operations is usually called Convolutional Backbone, and its role is to extract high-level features [1].

Usually, a CNN used to classify an image is composed of input, the convolutional backbone, and a classifier head. This last one is typically composed of fully connected artificial neural networks (ANN), which have several perceptrons connected among each other.

The process of finding the best weights of the neural network has two steps: a forward stage and a backward stage [27]. According to [27], the first step uses the current weights and biases of the network to process the input and calculate a prediction. Then this prediction is compared to the expected output (ground truth) with a function called loss. After determining the loss, the gradients of each parameter are updated in the backward stage using the chain rule, a method called backpropagation [9].

The objective of the training process is to minimize the loss function, which means that the outputs of the trained neural networks are similar to the ground truth. To carry out the training, the weights of the neural network need to be initialized, and the way they are set can impact the training time.

According to [65], two popular initialization methods are Glorot (a.k.a. Xavier initialization) [66] and He (a.k.a. Kaiming initialization) [67]: the first has as its primary goal achieve faster convergence and better accuracy by scaling the neural network weights so that the variance of the input is equal to the conflict of the output [65]; the second aims to achieve depth independent performance by modifying the scaling factor to account rectifier nonlinearities [65]. The weights of a neural network can also be initialized from a previously trained network, a technique that is known as transfer learning. [68] defines four types of transfer learning: instance-based, mapping-based, network-based, and adversarial-based.

To achieve convergence faster during the training process, some algorithms with adaptative learning rates can be used. In neural networks studies, these algorithms are usually gradient-based and are called optimizers [69]. Some examples of them are Stochastic Gradient Descend (SGD) [70], AdaGrad [71], Nesterov Accelerated Gradient (NAG) [72], Adaptative Moment Estimation (Adam) [73], Rectified Adam (RAdam) [74], Adaptative and Momental Bound (AdaMod) [75] and Adaptative Second Order (AdaHessian) [76].

Regarding loss functions, [77] summarizes some of the available ones that are usually chosen for semantic segmentation tasks. Among those, it is worth mentioning the ones that are commonly used in semantic segmentation papers: the Cross-Entropy (CE) [78], the Weighted Cross-Entropy (WCE) [79], the Dice [80], the IoU/Jaccard [81], the Tversky [82] and the Focal Tversky [83]. The mathematical formulation of each cited loss function is described respectively in the equations 10, 11, 12, 13, 14, and 15, where *N* is the number of pixels, g_i^c is the binary indicator of whether the class label c is correctly classified for pixel *i*, $s^c{}_i$ is the corresponding predicted probability, α and β are hyperparameters used to control the balance between false positives and false negatives, and γ is a coefficient in the interval [1,3].

Some metrics can be used to evaluate the quality of the trained neural networks. According to [84], overall accuracy (*OA*), precision, recall, and the F_1 index are helpful for evaluating the quality of the training, and they are defined by the following equations:

$$OA = \frac{TP + TN}{FP + FN} \tag{16}$$

$$precision = \frac{TP}{TP + FP} \tag{17}$$

$$recall = \frac{TP}{TP + FN} \tag{18}$$

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{19}$$

where TP, TN, FP, and FN are, respectively, the true positives, the true negatives, the false positives, and the false negatives.

According to [31], the Jaccard Index, also known as intersection over union (IoU), can be defined by:

$$IoU = J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{20}$$

where A e B are, respectively, the ground truth and the predicted data.

$$L_{CE} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} g_i^c \log s_i^c \tag{10}$$

$$L_{WCE} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} w_c g_i^c \log s_i^c \tag{11}$$

$$L_{Dice} = 1 - \frac{2 \sum_{i=1}^{N} \sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N} \sum_{c=1}^{C} g_i^{c2} + \sum_{i=1}^{N} \sum_{c=1}^{C} s_i^{c2}} \tag{12}$$

$$L_{IoU} = 1 - \frac{\sum_{i=1}^{N} \sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N} \sum_{c=1}^{C} \left(g_i^c + s_i^c - g_i^c s_i^c\right)} \tag{13}$$

$$L_{Tversky} = \frac{\sum_{i=1}^{N} \sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N} \sum_{c=1}^{C} \left(g_i^c s_i^c\right) + \alpha \sum_{i=1}^{N} \sum_{c=1} \left(1 - g_i^c\right) s_i^c + \beta \sum_{i=1}^{N} \sum_{c=1} g_i^c \left(1 - s_i^c\right)} \tag{14}$$

$$L_{FT} = \left(1 - L_{Tversky}\right)^{\frac{1}{\gamma}} \tag{15}$$

Also, according to [31], the mean intersection over union index (mIoU) can be defined by:

$$mIoU = \frac{1}{m}\sum\frac{A_{pred}\cap A_{true}}{A_{pred}\cup A_{true}} \quad (21)$$

where m is the number of expected classes, A_{pred} is the prediction set, and A_{true} is the ground truth set.

3.2. Convolutional Neural Networks Training Improving Techniques

Convolutional Neural Networks usually take a long time to train, even when using a GPU. This occurs due to the fact of the large number of weights that have to be adjusted in the process of backpropagation: the larger the number of parameters of the model, the longer it will take to train. This can be overcome using distributed training on several GPUs and increasing the batch size.

In addition, the time spent on the training process also depends on the number of samples that the training dataset has. On the one hand, if there are not enough images on the training dataset, the neural network will not" see" a significant number of patterns to learn and perform poorly on the training dataset. This below-average learning is known as underfitting. On the other hand, if the number of images is not high enough, the neural network can memorize the data and perform well on the training dataset, but poorly on the test dataset, known as overfit [64, 85].

Moreover, the performance on test datasets can be improved by using regularization techniques, which are defined by [64] as any modification made to a learning algorithm that is intended to reduce its generalization error but not its training error. Some examples of regularization techniques are weight decay, label smoothing, early stopping, dropout, batch normalization, and data augmentation. Each of these is described below:

- Weight decay (a.k.a. L2 Regularization) is a method that modifies the weights of a neural network in such a way that the loss to be minimized is added a penalty of the L_2 norm of the weights [64].
- Label smoothing [86, 64] is a technique that adds noise to the label, mitigating the effect of some incorrect label that the dataset may have. It also has the advantage of preventing the pursuit of hard probabilities without discouraging correct classification [64].
- Early stopping consists of stopping the training when the neural network stops learning, in other words, when the validation metrics stop improving [64].
- Dropout [87] is a technique used to reduce the dependency of some neurons on neural networks. At each training step, it is calculated a probability of the neuron to be shut down, and if it is larger than the set threshold, this element is turned off (outputs zero). This has a regularizing effect since it forces the network to learn patterns with other connected neurons.
- Batch Normalization [88] is a model reparameterization technique that introduces both additive and multiplicative noise on the hidden units at training time by normalizing the inputs to outputs with zero mean and unit variance [64].
- Data augmentation is a technique that uses image manipulation to create new training samples [64, 89]. Common data augmentation operations are random crop, random flip, and random color jitters. Furthermore, a novel data augmentation technique that has been recently employed in CS papers is Mixup [90], which consists of building synthetic images composed of a weighted sum of random pairs of the training data. According to [64, 89], data augmentation also has a regularizing effect, and it may contribute to avoid overfitting. One step further on data augmentation is using self-supervised techniques to learn from data the augmentation procedures that can achieve better metrics. As examples of such methods, we can cite AutoAugment [91], Faster AutoAugment [92], and RandAugment [93].

Furthermore, there is another approach to training optimization, which is the usage of Learning Rate Scheduling [94]. This technique changes the value of the learning rate according to some heuristic to try to improve the neural network accuracy and reduce training time [95, 96]. Some examples are Time Based Exponential Decay [97], Exponential Decay [98], Linear Warmup, Cosine Annealing [96], Cosine Power Annealing [99], and One-Cycle Learning Rate Scheduling Policy [100].

Finally, the last training improving technique that we will cover is Stochastic Weight Averaging (SWA) [101, 102], which is a procedure used to optimize the neural network that averages multiple points along the trajectory of Stochastic Gradient Descent (SGD), with specific learning rate procedures, that can be either cyclical or constant. The usage of this technique can help the optimizer to find a better optimization landscape, which might lead to better optimization results.

3.3. Main Convolutional Neural Network Backbones used on Semantic Segmentation Tasks

In this subsection, we will briefly present the key ideas regarding the main convolutional neural networks used to perform semantic segmentation tasks in RS. From our bibliographic research carried out in 2.1, we analyzed the results shown in figures 3 and 5, and then we identified key backbones to be explained in this section. The chosen backbones were AlexNet [22], ZFNet [23], GoogLeNet [24], VGG-19 [24], the ResNet family [103], Inception [86, 104], XCeption [105] and MobileNet [106, 107, 108]. From the bibliographic research done in Computer Sciences, we came across the following worth mentioning backbones: ResNeXt, ResNeSt, and EfficientNet.

According to [1, 109], convolutional neural networks (CNNs) were introduced by [9] and in 2012, [110] used them in a model called AlexNet to win the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [22]. According to [8], in 2013 and 2014, ILSVRC were also won by CNNs, with models respectively called ZFNet [23], GoogLeNet [24]. [1] define the architectures AlexNet [110], ZFNet [23] and VGG-19 [24] as Vintage Architectures.

In 2015, the family of architectures called ResNets [103] introduced skip connections to address the vanishing/exploding gradient [66, 111], which prevented deep neural networks from having a large number of layers. Due to this idea, deeper models were possible, and then the 2015's ILSVRC was won by a ResNet-152. The ResNet family has the ResNet blocks as its basic building blocks, a series of convolutions and activations stacked. There is a concatenation operation by the end of the block (also called skip connections) to preserve some of the input information.

To further push the boundary regarding the performance of the ResNet family-based algorithms, [86, 104] developed a family of architectures called Inception, which has as its basic block the inception block. Different from ResNet blocks that only concatenate the input of the block with the output, the inception block has several outputs: each output is the result of a different stacking of convolutions and pooling operations. Further advances on such idea were also proposed by the XCeption family [105] and the MobileNet family [106].

Thus, [112] evolved the idea of the Inception Block by proposing a backbone called ResNeXt: in this method, a cardinality value to the blocks is proposed, which widens the block with more branches of stacked convolutions, enabling further representation learning. Other backbone architectures that are worth mentioning are the SE-ResNet [50] and the ResNeSt [49]. The first method proposes the usage of an attention mechanism at the beginning and the end of the ResNet block, composing the Squeeze and Excite block, which performs dynamic channel-wise feature recalibration, to improve the representational power of the network. The latter method proposes the usage of Split-Attention Block, which adds the same idea of cardinality to the SE-Net-Block proposed by [50].

Recently there have been some breakthrough architectures using Neural Architecture Search (NAS) [113, 114, 115], which is a reinforcement learning technique to find out the best architecture to perform tasks on object detection and semantic segmentation [1]. Using NAS techniques, in late 2019, researchers at Google have created a series of backbones called EfficientNet [116]. In 2020, another group from Google had published a paper called EfficientDet: Scalable and Efficient Object Detection [117], in which they improved EfficientNets and proposed a weighted bi-directional feature pyramid network (BiFPN). According to [117], with these improvements, the research team achieved 4x smaller networks that used 13x fewer FLOPs, with a gain of 0.2% of mean average precision (mAP) of state-of-the-art mAP on the COCO dataset.

3.4. Main Convolutional Neural Network Architectures Used on Semantic Segmentation Tasks

In neural network applications, the convolutional backbone is often combined with other structures depending on the task that we want to perform. It can be used with a design such as fully convolutional layers to perform classification. In the case of semantic Segmentation, there are some approaches, as using naïve encoders and encoder-decoder structures [1]. There are also Generative Adversarial Networks (GAN) [39, 118, 119] and Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) [30] approaches to perform semantic segmentation tasks, but we will not cover those techniques in this paper. More information on those techniques can be found on [1, 30, 42].

Naïve decoders normally use a convolutional backbone and trained deconvolutional layers to perform the upsampling task to generate the segmentation mask, combined with some interpolation method such as bilinear. Some examples of this type of architecture are Fully Convolutional Networks (FCN) [120], DeepLabV1 [121], DeepLabV2 [122], ParseNet [123], PSPNet [124] and DeepLabV3 [125].

Encoder-decoder models, in contrast to naïve decoder, instead of using an interpolation method to upsample the feature maps, use a more complex decoder, with shortcuts or skip connections to maintain information from the encoder to the decoder and gradually perform the upsampling [1]. Some examples of this type of model are the DeconvNet [126], the SegNet [127], the U-Net [79], the U-Net++ [128], the DoubleU-Net [129], the MultiResUNet [130], the RefineNet [131] and the DeepLabV3+ [132]. The architecture of an encoder-decoder architecture called U-Net is shown in figure 6.

A novel type of encoder-decoder architecture is the HRNet (or High-Resolution Net) [133] and the HR-Net OCR[53], both of which are featured on top positions of the Cityscapes benchmark, as shown in table 1. This method aims to maintain high-resolution images at every stage of the process by combining different parallel chains of convolutions and strided convolutions. Object-Contextual Representations (OCR) is an attention mechanism [134] that considers the context of the considered pixel instead of it alone. OCR can be combined with different backbones such as ResNet-101 and Xception and different architectures such as DeepLabV3+ to improve segmentation results, as shown by [135]. When OCR is combined with HR-Net, we have the HR-Net OCR architecture.

Another type of attention mechanism that can be combined with HR-Net is the Polarized Self-Attention (PSA) [56], which has two main operations in its design: the polarized filtering and enhancement component. This type of attention mechanism not only looks at spatial features but also channel representations.

Finally, another worth mentioning set of techniques is the usage of EfficientNet backbones with Feature Pyramid Networks (FPN), combined with self-training techniques such as noisy student, which is a semi-supervised learning technique that improves the training results [57]. Table 1 shows that the best method on PASCAL VOC 2012 test dataset is the usage of EfficientNet trained with noisy student technique (a.k.a. EfficientNet-L2) with FPN architecture and Neural Architecture Search (NAS) [54]. On the other hand, the best model on PASCAL Context is the combination of a plain EfficientNet-B7 with an attention mechanism called Channelized Axial Attention (CAA) [55].

3.5. Applications on Remote Sensing and Examples of Available Datasets

Deep Learning (DL) plays an important role in nowadays science is particularly geosciences. There are several RS research papers such as [136], [137], and [138] that compare classical computer vision techniques to DL techniques, and they show that DL can achieve better accuracies.

DL-based techniques can solve several problems in Geosciences. Among those problems we can cite object detection [139, 140], hyperspectral image classification [10, 141], super-resolution [142, 143, 144], change detection [145, 146] and semantic segmentation.

Regarding Semantic Segmentation [84, 147, 148, 149], there are some use cases, such as building footprint extraction [11, 12, 150, 13, 14, 15, 16, 17, 18], road extraction [151, 152, 153] and land use and land cover (LULC) analysis [154, 155].

To train neural networks that can solve LULC problems, data from the ISPRS Potsdam and Vaihingen [156, 157] can be used. This is a dataset with airborne photogrammetric imagery of Potsdam, covering six classes (impervious surfaces, building, low vegetation, tree, car, and clutter/background).

Moreover, to perform training of deep convolutional neural networks that can extract building footprints, some of the open datasets available online are listed below, and the details are shown in table 2:

- SpaceNet [158, 159]: dataset with satellite imagery of the following cities: Rio de Janeiro, Las Vegas, Paris, Khartoum, and Shanghai.
- Massachusetts [160]: dataset with satellite imagery of the city of Boston.
- WHU building [161]: dataset with airborne photogrammetric imagery of New Zealand.
- INRIA aerial [162]: dataset with satellite imagery from the following cities: Austin, Chicago, Kitsap County, Western Tyrol, and Vienna.
- LandCover.ai [163]: dataset with satellite imagery of Poland.
- AIRS [164]: dataset with satellite imagery of Christchurch City in New Zealand.
- CrowdAI [165]: a simplified version of the SpaceNet Dataset, with only RGB images.

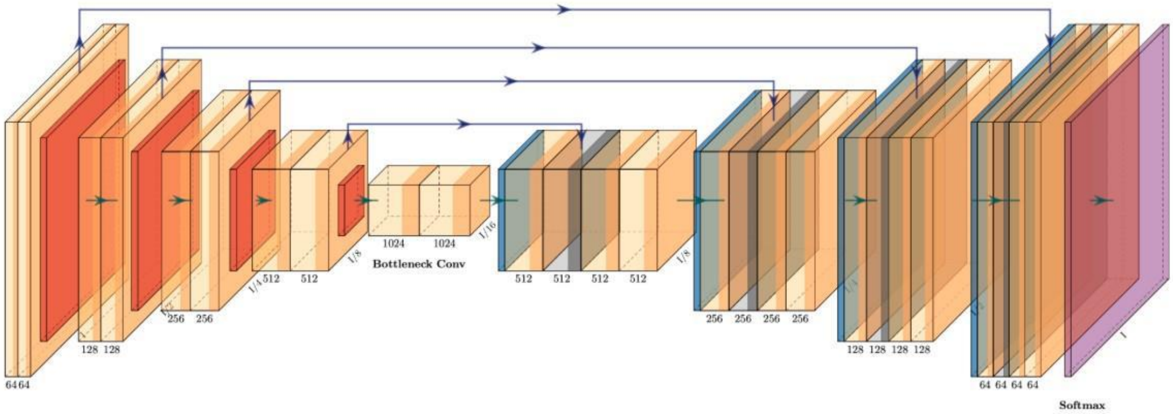


Fig. 6: Basic structure of a U-Net. Figure built using https://github.com/HarisIqbal88/PlotNeuralNet.

Table 2: Comparison between building footprint datasets

Dataset	# of buildings	# of tiles	Tile Size	Spatial Resolution
LandCover.ai	12,788	41	33 tiles with the size 9000 x 9500 px and eight tiles with size 4200 x 4700 px	25cm and 50 cm
INRIA	216,418	360	5000 x 5000 px	30 cm
Massachusetts Buildings	310,425	151	1500 x 1500 px	1 m
Spacenet	462,091	17,533	512 x 512 px	35 cm
WHU building dataset	220,000	25,577	512 x 512 px	7.5 cm and 2.7 cm
AIRS	220,000	1,047	10,000 x 10,000 px	7.5 cm
CrowdAI	Unknown	280,741 training images, 60,317 validation images and 60,697 test images	300 x 300 px	Unknown

3.6. Available Frameworks and Tools

The two most famous deep learning frameworks are Tensorflow [166] and PyTorch [167]. Both are open source, have large communities, are very well documented, and have outstanding performance. Tensorflow has an underlying library called Keras [168], enabling a higher level and more readable code. On the PyTorch side, PyTorch Lightning [169], FastAI [170], and Catalyst [171], among others, are frameworks that provide similar improvements given by Keras.

Considering segmentation models tools openly available, there are two frameworks developed in Python that use Tensorflow and PyTorch, respectively segmentation models [172] and segmentation models PyTorch [173]. To train segmentation models without coding skills, users can build a JSON file with the parameters of the training and use a Python package called segmentation models trainer [174], which was built using Tensorflow, Keras, and segmentation models. [175] has also created a training framework using PyTorch and PyTorch Lightning called PyTorch segmentation models trainer, which instead of using a JSON to fill the hyperparameters, uses a YAML file using configuration composition, which enables users to reuse settings. To build training masks from vector data, a QGIS [176] plugin called DeepLearningTools [177] can be used.

There are also tools to help to build and to inspect datasets, such as FiftyOne [178]. With this tool, data scientists can visualize the labels overlapped to the images and calculate image similarity indexes to assess the quality of the dataset and identify missing labels.

Concerning data augmentation, each library has built-in operations. As external options, we can cite Albumentations [179], a Python package that is framework agnostic and works only on CPU. Another option on the PyTorch ecosystem is Kornia [180], a package that works on either CPU or GPU.

IV. CONCLUSION

In this paper, we presented the SOTA of Semantic Segmentation in Remote Sensing, an ever-growing field of research, with an almost exponential increase in the number of publications, as shown in section 2.1. We identified that the most used backbones on RS tasks are the ResNet family, VGG-16, Inception-V3, and AlexNet. Furthermore, we identified that the most famous architectures used in RS are the U-Net, DeepLabV3+, FCN, and SegNet. We also briefly showed the main theories, algorithms, and neural networks architectures and backbones.

This paper has also briefly presented how convolutional neural networks work and the techniques used for training such structures, like weight initialization, popular optimizers, some of the loss functions available, and the often-used metrics in RS papers. We also showed some of the existing regularizing techniques such as weight decay, label smoothing, early stopping, dropout, batch normalization, and data augmentation.

Then, we also presented some learning rate scheduling methods and stochastic weight averaging. We also listed the most famous backbones and architectures found on the RS papers surveyed and presented some applications of such techniques on RS. We also showed some available datasets and popular frameworks and packages to train deep learning convolutional neural networks.

There are many research papers in CS that propose several neural architectures, and some have been used in RS applications. Deep Learning is an ever-growing field, and in 2020 there have been many promising and exciting new backbones, such as the EfficientNet family, the ResNeSt-269 [49], and the SE-ResNet family [50].

Moreover, we have identified a research opportunity in RS to combine the mentioned backbones with popular architectures such as U-Net, FPNs, and PSPNet. Another research opportunity is the usage of HRNet-OCR [53], HRNetV2-OCR+PSA [56], EfficientNet-B7+CAA [55], and EfficientNet-L2+NAS-FPN [54], which are in the leader board of Papers With Code [21], but was not observed in the surveyed papers regarding remote sensing applications.

In addition, another research opportunity that we identified is to perform an extensive comparison of the accuracy of trained models with several combinations of neural networks architectures and backbones to define the best method to extract information from very-high remote sensing images. We can also highlight other research opportunities, such as determining the best loss function to be used in training and the best inference method to improve validation data accuracy. The suggested loss function for such a study is the Focal Tversky [83] since it handles class imbalance problems, a common problem in remote sensing datasets, especially building footprint extraction datasets.

Additionally, even though new optimizers such as RAdam, AdaMod, and AdaHessian have been proposed, few papers in remote sensing have tested them. The same principle can be applied to activation functions such as Leaky-ReLU, ELU, SELU, GELU, and Mish. So, we also identify research opportunities of the influence of optimizers and activation functions in the training time and the test metric scores.

Finally, other aspects that we did not find in the surveyed papers and that can be researched is the usage of stochastic weight averaging [101, 102], novel augmentation techniques such as Mixup [90], AutoAugment [91], Faster AutoAugment [92] and RandAugment [93].

ACKNOWLEDGEMENTS

We would like to acknowledge the Brazilian Army Geographic Service and the Institute of Geosciences of the University of Brasília for their support in our research.

REFERENCES

[1] Thorsten Hoeser and Claudia Kuenzer, “Object detection and image segmentation with deep learning on Earth observation data: A review-part I: Evolution and recent trends,” *Remote Sensing*, vol. 12, no. 10, 2020.

[2] Thomas Blaschke, Geoffrey JHay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, Freek Van der Meer, Harald Van der Werff, Frieke Van Coillie, et al., “Geographic object-based image analysis–towards a new paradigm,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 87, pp. 180–191, 2014.

[3] Gang Li and Youchuan Wan, “Adaptive watershed segmentation of remote sensing image based on wavelet transform and fractal dimension,” in *Proceedings of the 2011, International Conference on Informatics, Cybernetics, and Computer Engineering (ICCE2011) November 19–20, 2011, Melbourne, Australia*. Springer, 2011, pp. 57–67.

[4] Hua Jiang, GuiLin Xu, and Jing Qin, “Research on adaptive model of remote sensing image segmentation based on graph theory,” in *Computer Application and*

System Modeling (ICCASM), 2010 International Conference on. IEEE, 2010, vol. 6, pp. V6–445.

[5] Bo Peng, Xingzheng Wang, and Yan Yang, "Region based exemplar references for image segmentation evaluation," *IEEE Signal Processing Letters*, vol. 23, no. 4, pp. 459–462, 2016.

[6] MadodomziMafanya, Philemon Tsele, Joel Botai, PhetoleManyama, Barend Swart, and Thabang Monate, "Evaluating pixel and object based image classification techniques for mapping plant invasions from uav derived aerial imagery: Harrisiapomanensis as a case study," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 129, pp. 1–11, 2017.

[7] Jim X. Chen, "The Evolution of Computing: AlphaGo," *Computing in Science and Engineering*, vol. 18, no. 4, pp. 4–7, 2016.

[8] John E. Ball, Derek T. Anderson, and Chee Seng Chan, "Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community," *Journal of Applied Remote Sensing*, vol. 11, no. 04, pp. 1, 2017.

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, dec 1989.

[10] Xiao Xiang Zhu, DevisTuia, LichaoMou, Gui Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer, "Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources," dec 2017.

[11] K. Bittner, S. Cui, and P. Reinartz, "Building extraction from remote sensing data using fully convolutional networks," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 42, no. 1W1, pp. 481–486, 2017.

[12] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L. Waslander, "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 42–55, jan 2019.

[13] Shengsheng Wang, Xiaowei Hou, and Xin Zhao, "Automatic Building Extraction from High-Resolution Aerial Imagery via Fully Convolutional EncoderDecoder Network with Non-Local Block," *IEEE Access*, vol. 8, pp. 7313–7322, 2020.

[14] Kang Zhao, Muhammad Kamran, and Gunho Sohn, "Boundary Regularized Building Footprint Extraction From Satellite Images Using Deep Neural Network," 2020.

[15] Wei Guo, Weihong Li, Weiguo Gong, and Jinkai Cui, "Extended Feature Pyramid Network with Adaptive Scale Training Strategy and Anchors for Object Detection in Aerial Images," *Remote Sensing*, vol. 12, no. 5, pp. 784, mar 2020.

[16] Guang Yang, Qian Zhang, and Guixu Zhang, "EANet: Edge-Aware Network for the Extraction of Buildings from Aerial Images," *Remote Sensing*, vol. 12, no. 13, pp. 2161, 2020.

[17] L. Hang and G. Y. Cai, "Cnn Based Detection of Building Roofs From High Resolution Satellite Images," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-3/W10, no. November 2019, pp. 187–192, 2020.

[18] Jingjing Ma, Linlin Wu, Xu Tang, Fang Liu, Xiangrong Zhang, and Licheng Jiao, "Building Extraction of Aerial Images by a Global and Multi-Scale EncoderDecoder Network," *Remote Sensing*, vol. 12, no. 15, pp. 2350, 2020.

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3431–3440.

[20] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross' Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[21] Facebook, "Papers with code," 2018.

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211– 252, 2015.

[23] Matthew D. Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8689 LNCS, no. PART 1, pp. 818–833, 2014.

[24] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 1–9, 2015.

[25] Thorsten Hoeser, Felix Bachofer, and Claudia Kuenzer, "Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review—Part II: Applications," *Remote Sensing*, vol. 12, no. 18, pp. 3053, 2020.

[26] Jurgen Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[27] Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[28] Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International*

Journal of Automation and Computing, vol. 14, no. 2, pp. 119–135, 2017.

[29] Yanming Guo, Yu Liu, Theodoros Georgiou, and Michael S. Lew, “A review of semantic segmentation using deep neural networks,” *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 87–93, 2018.

[30] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, Pablo MartinezGonzalez, and Jose Garcia-Rodriguez, “A survey on deep learning techniques for image and video semantic segmentation,” *Applied Soft Computing Journal*, vol. 70, pp. 41–65, 2018.

[31] Hongshan Yu, Zhengeng Yang, Lei Tan, Yaonan Wang, Wei Sun, Mingui Sun, and Yandong Tang, “Methods and datasets on semantic segmentation: A review,” *Neurocomputing*, vol. 304, pp. 82–103, 2018.

[32] Arne Schumann, Lars Sommer, KrassimirValev, and Jurgen Beyerer, “A systematic evaluation of recent deep learning architectures for fine-grained vehicle classification,” p. 1, 2018.

[33] Farhana Sultana, Abu Sufian, and Paramartha Dutta, “Advancements in image classification using convolutional neural network,” *Proceedings - 2018 4th IEEE International Conference on Research in Computational Intelligence and Communication Networks, ICRCICN 2018*, pp. 122–129, 2018.

[34] Md ZahangirAlom, Tarek M. Taha, Chris Yakopcic, Stefan Westberg, PahedingSidike, MstShamima Nasrin, Mahmudul Hasan, Brian C. Van Essen, Abdul A.S.Awwal, and Vijayan K. Asari, “A state-of-the-art survey on deep learning theory and architectures,” *Electronics (Switzerland)*, vol. 8, no. 3, 2019.

[35] Swarnendu Ghosh, Nibaran Das, Ishita Das, and Ujjwal Maulik, “Understanding deep learning techniques for image segmentation,” *ACM Computing Surveys*, vol. 52, no. 4, 2019.

[36] Xiaolong Liu, Zhidong Deng, and Yuhan Yang, “Recent progress in semantic image segmentation,” *Artificial Intelligence Review*, 2018.

[37] Asifullah Khan, AnabiaSohail, UmmeZahoora, and Aqsa Saeed Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artificial Intelligence Review*, pp. 1–70, 2020.

[38] Shijie Hao, Yuan Zhou, and Yanrong Guo, “A Brief Survey on Semantic Segmentation with Deep Learning,” *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[39] Shervin Minaee, Yuri Boykov, FatihPorikli, AntonioPlaza, Nasser Kehtarnavaz, and Demetri Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” pp. 1–23, 2020.

[40] SaeidAsgariTaghanaki, Kumar Abhishek, Joseph PaulCohen, Julien Cohen-Adad, and Ghassan Hamarneh, “Deep semantic segmentation of natural and medical images: a review,” *Artificial Intelligence Review*, 2020.

[41] Yuzhu Ji, Haijun Zhang, Zhao Zhang, and Ming Liu, “Cnn-based encoder-decoder networks for salient object detection: A comprehensive review and recent advances,” *Information Sciences*, vol. 546, pp. 835–857, 2021.

[42] Shijie Hao, Yuan Zhou, and Yanrong Guo, “A brief survey on semantic segmentation with deep learning,” *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[43] Baojun Li, Shun Liu,Weichao Xu, and Wei Qiu, “Real-time object detection and semantic segmentation for autonomous driving,” in *MIPPR 2017: Automatic Target Recognition and Navigation*. International Society for Optics and Photonics, 2018, vol. 10608, p. 106080P.

[44] Yu-Ho Tseng and Shau-Shiun Jan, “Combination of computer vision detection and segmentation for autonomous driving,” in *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2018, pp. 1047–1052.

[45] Fabian Flohr, DariuGavrila, et al., “Pedcut: an iterative framework for pedestrian segmentation combining shape models and multiple data cues.,” in *BMVC*, 2013.

[46] Garrick Brazil, Xi Yin, and Xiaoming Liu, “Illuminating pedestrians via simultaneous detection & segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4950–4959.

[47] Xiaofeng Zhu, Heung-Il Suk, Seong-Whan Lee, and Dinggang Shen, “Subspace regularized sparse multitask learning for multiclass neurodegenerative disease identification,” *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 3, pp. 607–618, 2015.

[48] Xiaofeng Zhu, Heung-Il Suk, and Dinggang Shen, “A novel matrix-similarity based loss function for joint regression and classification in ad diagnosis,” *NeuroImage*, vol. 100, pp. 91–105, 2014.

[49] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, Mu Li, and Alexander Smola, “ResNeSt: Split-Attention Networks,” *arXiv*, 2020.

[50] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu, “Squeeze-and-Excitation Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.

[51] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, RodrigoBenenson, Uwe Franke, Stefan Roth, and Bernt Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[52] Longlong Jing and Yingli Tian, “Self-supervised visual feature learning with deep neural networks: A survey,” 2019.

[53] Andrew Tao, Karan Sapra, and Bryan Catanzaro, “Hierarchical Multi-Scale Attention for Semantic Segmentation,” 2020.

[54] Barret Zoph, GolnazGhiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le, “Rethinking pre-training and self-training,” 2020.

[55] Ye Huang, Wenjing Jia, Xiangjian He, Liu Liu, Yuxin Li, and Dacheng Tao, “Channelized axial attention for semantic segmentation,” 2021.

[56] Huajun Liu, Fuqiang Liu, Xinyi Fan, and Dong Huang, “Polarized self-attention: Towards high-quality pixelwise regression,” 2021.

[57] QizheXie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le, “Self-training with noisy student improves imagenet classification,” 2020.

[58] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi, “Convolutional neural networks: an overview and application in radiology,” *Insights into Imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[59] Andrew L Maas, Awni Y Hannun, and Andrew YNg, “Rectifier nonlinearities improve neural network acoustic models,” *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, vol. 28, 2013.

[60] Djork Arne Clevert, Thomas Unterthiner, and Sepp’ Hochreiter, “Fast and accurate deep network learning by exponential linear units (ELUs),” *4th International Conference on Learning Representations, ICLR 2016 Conference Track Proceedings*, pp. 1–14, 2016.

[61] Gunter¨ Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter, “Self-normalizing neural networks,” *Advances in Neural Information Processing Systems*, vol. 2017-December, pp. 972–981, 2017.

[62] Dan Hendrycks and Kevin Gimpel, “Gaussian Error Linear Units (GELUs),” pp. 1–9, 2016.

[63] DigantaMisra, “Mish: A self-regularized nonmonotonic neural activation function,” *arXiv*, 2019.

[64] Ian Goodfellow, YoshuaBengio, and Aaron Courville, *Deep learning*, MIT press, 2016.

[65] Meenal V. Narkhede, Prashant P. Bartakke, and Mukul S. Sutaone, *A review on weight initialization strategies for neural networks*, Number 0123456789. Springer Netherlands, 2021.

[66] Xavier Glorot and YoshuaBengio, “Understanding the difficulty of training deep feedforward neural networks,” *Journal of Machine Learning Research*, vol. 9, pp. 249–256, 2010.

[67] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Delving deep into rectifiers: Surpassing humanlevel performance on imagenet classification,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[68] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu, “A survey on deep transfer learning,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11141 LNCS, pp. 270–279, 2018.

[69] Ian Goodfellow, YoshuaBengio, and Aaron Courville, *Deep Learning*, 2017.

[70] Leon Bottou,’ “Large-Scale Machine Learning with Stochastic Gradient Descent,” *Proceedings of COMPSTAT’2010*, 2010.

[71] John Duchi, Elad Hazan, and Yoram Singer, “Adaptive subgradient methods for online learning and stochastic optimization,” *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2011.

[72] Y. NESTEROV, “A method for solving the convex programming problem with convergence rate O(1/kˆ2),” 1983.

[73] Diederik P Kingma and Jimmy Lei Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–15.

[74] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han, “On the Variance of the Adaptive Learning Rate and Beyond,” pp. 1–14, 2019.

[75] Jianbang Ding, Xuancheng Ren, Ruixuan Luo, and Xu Sun, “An adaptive and momental bound method for stochastic learning,” *arXiv*, 2019.

[76] Zhewei Yao, Amir Gholami, Sheng Shen, Kurt Keutzer, and Michael W. Mahoney, “ADAHESSIAN: An Adaptive Second Order Optimizer for Machine Learning,” *arXiv*, pp. 1–20, 2020.

[77] Jun Ma, “Segmentation Loss Odyssey,” 2020.

[78] Kevin P Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.

[79] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vol. 9351, pp. 234–241.

[80] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” in *2016 fourth international conference on 3D vision (3DV)*. IEEE, 2016, pp. 565–571.

[81] Md Atiqur Rahman and Yang Wang, “Optimizing intersection-over-union in deep neural networks for image segmentation,” in *International symposium on visual computing*. Springer, 2016, pp. 234–244.

[82] SeyedRaein Hashemi, SeyedSadeghMohseni Salehi, Deniz Erdogmus, Sanjay P Prabhu, Simon K Warfield, and Ali Gholipour, “Asymmetric loss functionsand deep densely-connected networks for highlyimbalanced medical image segmentation: Application to multiple sclerosis lesion detection,” *IEEE Access*, vol. 7, pp. 1721–1735, 2018.

[83] Nabila Abraham and NaimulMefraz Khan, “A novel focal tversky loss function with improved attention unet for lesion segmentation,” 2018.

[84] Foivos I. Diakogiannis, Franc¸ois Waldner, Peter Caccetta, and Chen Wu, “ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data,” 2019.

[85] Zeju Li, Konstantinos Kamnitsas, and Ben Glocker, “Overfitting of neural nets under class imbalance: Analysis and improvements for segmentation,” in

International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, 2019, pp. 402–410.

[86] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[87] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[88] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015.

[89] Connor Shorten and Taghi M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, 2019.

[90] Hongyi Zhang, MoustaphaCisse, Yann N. Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," 2018.

[91] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le, "Autoaugment: Learning augmentation policies from data," 2019.

[92] Ryuichiro Hataya, Jan Zdenek, KazukiYoshizoe, and Hideki Nakayama, "Faster AutoAugment: Learning Augmentation Strategies Using Backpropagation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12370 LNCS, pp. 1–16, 2020.

[93] Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," 2019.

[94] Jieun Park, Dokkyun Yi, and Sangmin Ji, "A novel learning rate schedule in optimization for neural networks and it's convergence," *Symmetry*, vol. 12, no. 4, 2020.

[95] Leslie N. Smith, "Cyclical learning rates for training neural networks," *Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017*, no. April 2015, pp. 464–472, 2017.

[96] Ilya Loshchilov and Frank Hutter, "Sgdr: Stochastic gradient descent with warm restarts," 2017.

[97] YuriiNesterov, "Introduction to convex optimization: A basic course," 2004.

[98] Zhiyuan Li and Sanjeev Arora, "An exponential learning rate schedule for deep learning," *arXiv preprint arXiv:1910.07454*, 2019.

[99] Andrew Hundt, Varun Jain, and Gregory D. Hager, "sharpdarts: Faster and more accurate differentiable architecture search," 2019.

[100] Leslie N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay," 2018.

[101] Pavel Izmailov, DmitriiPodoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson, "Averaging weights leads to wider optima and better generalization," *arXiv preprint arXiv:1803.05407*, 2018.

[102] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson, "There are many consistent explanations of unlabeled data: Why you should average," *arXiv preprint arXiv:1806.05594*, 2018.

[103] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and JianSun, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 770–778, 2016.

[104] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," 2016.

[105] Franc¸ois Chollet, "Xception: Deep learning with depthwise separable convolutions," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 1800–1807, 2017.

[106] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[107] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[108] Andrew Howard, Mark Sandler, Grace Chu, LiangChieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al., "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1314–1324.

[109] Dan Cires¸anCires¸an, Ueli Meier, and Jurgen Schmid-¨ huber, "Multi-column Deep Neural Networks for Image Classification," Tech. Rep., 2012.

[110] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *COMMUNICATIONS OF THE ACM*, vol. 60, no. 6, 2017.

[111] YoshuaBengio, Patrice Simard, and Paolo Frasconi, "Learning Long-Term Dependencies with Gradient Descent is Difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[112] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen' Tu, and Kaiming He, "Aggregated residual transformations for deep neural networks," *Proceedings 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5987– 5995, 2017.

[113] Barret Zoph and Quoc V. Le, "Neural architecture search with reinforcement learning," *5th International Conference on Learning Representations, ICLR 2017 Conference Track Proceedings*, pp. 1–16, 2017.

[114] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan Yuille, and Li FeiFei, "Auto-

DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation," Tech. Rep.

[115] GolnazGhaisi, Tsung-Yi Lin, Ruoming Pang Quoc, and V Le Google Brain, "NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection," Tech. Rep.

[116] Mingxing Tan and Quoc V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," Tech. Rep., 2019.

[117] Mingxing Tan, Ruoming Pang, and Quoc V Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10781– 10790.

[118] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, SherjilOzair, Aaron Courville, and YoshuaBengio, "Sparse generative adversarial network," *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, pp. 3063–3071, 2019.

[119] Clint Sebastian, Raffaele Imbriaco, Egor Bondarev, and Peter H. N. de With, "Adversarial Loss for Semantic Segmentation of Aerial Imagery," 2020.

[120] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation," Tech. Rep.

[121] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," *ICLR 2015*, dec 2014.

[122] Liang Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, apr 2018.

[123] Wei Liu, Andrew Rabinovich, and Alexander C. Berg, "ParseNet: Looking Wider to See Better," 2015.

[124] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, "Pyramid scene parsing network," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 6230–6239, 2017.

[125] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," Tech. Rep., 2017.

[126] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han, "Learning deconvolution network for semantic segmentation," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 International Conference on Computer Vision, ICCV 2015, pp. 1520–1528, 2015.

[127] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla, "Segnet: A deep convolutional encoderdecoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[128] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, NimaTajbakhsh, and Jianming Liang, "Unet++: A nested u-net architecture for medical image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11045 LNCS, pp. 3–11, 2018.

[129] Debesh Jha, Michael A. Riegler, Dag Johansen, Pal˚ Halvorsen, and Havard D. Johansen, "DoubleU-Net: A˚ Deep Convolutional Neural Network for Medical Image Segmentation," jun 2020.

[130] Nabil Ibtehaz and M Sohel Rahman, "MultiResUNet : Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Networks*, vol. 121, pp. 74–87, 2020.

[131] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," *Proceedings 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5168–5177, 2017.

[132] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "EncoderDecoder with Atrous Separable Convolution for Semantic Image Segmentation," Tech. Rep.

[133] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao, "Deep high-resolution representation learning for visual recognition," 2020.

[134] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and IlliaPolosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[135] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang, "Segmentation transformer: Objectcontextual representations for semantic segmentation," 2021.

[136] Mabel Ortega Adarme, Raul Queiroz Feitosa, Patrick NigriNigriHapp, Claudio Aparecido De Almeida, and Alessandra Rodrigues Gomes, "Evaluation of Deep Learning Techniques for Deforestation Detection in the Brazilian Amazon and Cerrado Biomes From Remote Sensing Imagery," *Remote Sensing*, vol. 12, no. 6, pp. 910, mar 2020.

[137] NicholusMboga, Stefanos Georganos, Tais Grippa, Moritz Lennert, Sabine Vanhuysse, and Eleonore' Wolff, "Fully convolutional networks and geographic object-based image analysis for the classification of VHR imagery," *Remote Sensing*, vol. 11, no. 5, 2019.

[138] Emilio Guirado, SihamTabik, Domingo AlcarazSegura, Javier Cabello, and Francisco Herrera, "Deeplearning Versus OBIA for scattered shrub detection with Google Earth Imagery: Ziziphus lotus as case study," *Remote Sensing*, vol. 9, no. 12, pp. 1–22, 2017.

[139] Musyarofah, Valentina Schmidt, and Martin Kada, "Object detection of aerial image using mask-region convolutional neural network (mask R-CNN)," in *IOP*

Conference Series: Earth and Environmental Science. jul 2020, vol. 500, Institute of Physics Publishing.

[140] Kun Li, Xiangyun Hu, Huiwei Jiang, Zhen Shu, and Mi Zhang, "Attention-Guided Multi-Scale Segmentation Neural Network for Interactive Extraction of Region Objects from High-Resolution Satellite Imagery," 2020.

[141] Hyperspectral Data, Yushi Chen, Zhouhan Lin, YushiChen, Zhouhan Lin, Xing Zhao, and Student Member, "Deep Learning-Based Classification of Hyperspectral Data," vol. 7, no. June 2014, pp. 1–14, 2015.

[142] Charis Lanaras, Jose' Bioucas-Dias, SilvanoGalliani, Emmanuel Baltsavias, Konrad Schindler, Remote Sensing, and Eth Zurich, "Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network," 2018.

[143] Mengjiao Qin, Sebastien' Mavromatis, Linshu Hu, Feng Zhang, Renyi Liu, Jean Sequeira, and Zhenhong Du, "Remote Sensing Single-Image Resolution Improvement Using A Deep Gradient-Aware Network with Image-Specific Enhancement," *Remote Sensing*, vol. 12, no. 5, pp. 758, feb 2020.

[144] Francesco Salvetti, Vittorio Mazzia, Aleem Khaliq, and Marcello Chiaberge, "Multi-Image Super Resolution of Remotely Sensed Images Using Residual Attention Deep Neural Networks," *Remote Sensing*, vol. 12, no. 14, pp. 2207, jul 2020.

[145] Yi Liu, Chao Pang, Zongqian Zhan, Xiaomeng Zhang, and Xue Yang, "Building Change Detection for Remote Sensing Images Using a Dual Task Constrained Deep Siamese Convolutional Network Model," 2019.

[146] Qing Wang, Xiaodong Zhang, Guanzhou Chen, Fan Dai, Yuanfu Gong, and Kun Zhu, "Change detection based on Faster R-CNN for high-resolution remote sensing images," *Remote Sensing Letters*, vol. 9, no. 10, pp. 923–932, oct 2018.

[147] Liangpei Zhang, Lefei Zhang, and Bo Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[148] Ronald Kemker, Carl Salvaggio, and Christopher Kanan, "Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 145, pp. 60–77, 2018.

[149] Hui Yang, Penghai Wu, Xuedong Yao, Yanlan Wu, Biao Wang, and Yongyang Xu, "Building extraction in very high-resolution imagery by dense-attention networks," *Remote Sensing*, vol. 10, no. 11, pp. 1–16, 2018.

[150] Lili Zhang, Jisen Wu, Yu Fan, Hongmin Gao, and Yehong Shao, "An efficient building extraction method from high spatial resolution remote sensing images based on improved mask R-CNN," *Sensors (Switzerland)*, vol. 20, no. 5, pp. 1–13, 2020.

[151] Yiheng Zhang, ZhaofanQiu, Ting Yao, Dong Liu, and Tao Mei, "Fully Convolutional Adaptation Networks for Semantic Segmentation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 6810–6818, 2018.

[152] Ming Wu, Chuang Zhang, Jiaming Liu, Lichen Zhou, and Xiaoqi Li, "Towards Accurate High Resolution Satellite Image Semantic Segmentation," *IEEE Access*, vol. 7, pp. 55609–55619, 2019.

[153] Renbao Lian and Liqin Huang, "DeepWindow: Sliding Window Based on Deep Learning for Road Extraction from Remote Sensing Images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, no. d, pp. 1–1, 2020.

[154] Shahab EddinJozdani, Brian Alan Johnson, and Dongmei Chen, "Comparing Deep Neural Networks, Ensemble Classifiers , and Support Vector Machine Algorithms," *Remote Sensing*, vol. 11, no. 1, pp. 1–24, 2019.

[155] Manuel Carranza-Garc´ıa, Jorge Garc´ıa-Gutierrez, and' Jose C. Riquelme,' "A framework for evaluating land use and land cover classification using convolutional neural networks," *Remote Sensing*, vol. 11, no. 3, 2019. [156] ISPRS, "2d semantic labeling contest," 2012.

[157] Ahram Song and Jaewan Choi, "Fully Convolutional Networks with Multiscale 3D Filters and Transfer Learning for Change Detection in High Spatial Resolution Satellite Images," *Remote Sensing*, vol. 12, no. 5, pp. 799, mar 2020.

[158] The SpaceNet Catalog, "Spacenet on amazon web services (aws)," 2018.

[159] Adam Van Etten, Dave Lindenbaum, and Todd Bacastow, "SpaceNet: A remote sensing dataset and challenge series," *arXiv*, 2018.

[160] Volodymyr Mnih, "Machine Learning for Aerial Image Labeling," *PhD Thesis*, p. 109, 2013.

[161] Yang Long, Gui Song Xia, Shengyang Li, Wen Yang, Michael Ying Yang, Xiao Xiang Zhu, Liangpei Zhang, and Deren Li, "DiRS: On creating benchmark datasets for remote sensing image interpretation," *arXiv*, pp. 1–22, 2020.

[162] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez, "Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2017-July, pp. 3226–3229, 2017.

[163] Adrian Boguszewski, Dominik Batorski, Natalia Ziemba-Jankowska, Anna Zambrzycka, and Tomasz Dziedzic, "LandCover.ai: Dataset for Automatic Mapping of Buildings, Woodlands and Water from Aerial Imagery," pp. 1–14, 2020.

[164] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L. Waslander, "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, no. November 2018, pp. 42–55, 2019.

[165] Sharada Prasanna Mohanty, Jakub Czakon, Kamil A Kaczmarek, Andrzej Pyskir, Piotr Tarasiewicz, Saket Kunwar, Janick Rohrbach, Dave Luo, Manjunath Prasad,

Sascha Fleer, et al., “Deep learning for understanding satellite imagery: An experimental survey,” *Frontiers in Artificial Intelligence*, vol. 3, 2020.

[166] Mart´ın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, SanjayGhemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, RafalJozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mane, Rajat Monga, Sherry Moore,’ Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin’ Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015, Software available from tensorflow.org.

[167] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, AlykhanTejani, SasankChilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and SoumithChintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alche-Buc, E. Fox, and R. Garnett, Eds., pp. 8024–’ 8035. Curran Associates, Inc., 2019.

[168] Francois Chollet et al., “Keras,” https://github. com/fchollet/keras, 2015.

[169] WA Falcon, “Pytorch lightning,” *GitHub. Note:https://github.com/PyTorchLightning/pytorchlightning*, vol. 3, 2019.

[170] Jeremy Howard and Sylvain Gugger, “Fastai: A layered api for deep learning,” *Information (Switzerland)*, vol. 11, no. 2, pp. 1–26, 2020.

[171] Sergey Kolesnikov, “Accelerated deep learning rd,” https://github.com/catalyst-team/ catalyst, 2018.

[172] Pavel Yakubovskiy, “Segmentation models,” https://github.com/qubvel/ segmentation_models, 2019.

[173] Pavel Yakubovskiy, “Segmentation models pytorch,” https://github.com/qubvel/ segmentation_models.pytorch, 2020.

[174] Philipe Borba, “phborba/segmentation models trainer: First Release,” sep 2020.

[175] Philipe Borba, “phborba/pytorch segmentation models trainer: Version 0.8.0,” July 2021.

[176] QGIS Development Team, *QGIS Geographic Information System*, Open-Source Geospatial Foundation, 2009.

[177] Philipe Borba, “phborba/DeepLearningTools: First release,” oct 2020.

[178] B. E. Moore and J. J. Corso, “Fiftyone,” *GitHub. Note: https://github.com/voxel51/fiftyone*, 2020.

[179] E. Khvedchenya V. I. Iglovikov A. Buslaev, A. Parinov and A. A. Kalinin, “Albumentations: fast and flexible image augmentations,” *ArXiv e-prints*, 2018.

[180] J. Shi D. Ponsa F. Moreno-Noguer E. Riba, D. Mishkin, and G. Bradski, “A survey on kornia: an open-source differentiable computer vision library for PyTorch,” 2020.

4.2 Discussão do Artigo de Revisão

Na subseção 4.1, foi apresentado o artigo de título ”*A Review of Remote Sensing Applications on Very High-Resolution Imagery Using Deep Learning-Based Semantic Segmentation Techniques*”, publicado no International Journal of Advanced Engineering Research and Science (IJAERS) (BORBA et al., 2021a). Neste artigo, foram apontados os conceitos básicos necessários para o entendimento de técnicas de segmentação semântica, a saber: principais arquiteturas e *backbones*, alguns *optimizers*, funções de ativação e funções de perda (*losses*), alguns conjuntos de dados (*datasets*), técnicas utilizadas no treinamento de redes neurais, algumas aplicações em sensoriamento remoto e *frameworks* disponíveis.

Da elaboração do referido artigo, pode-se depreender que há diversos trabalhos com resultados relevantes que utilizam arquiteturas como a U-Net (RONNEBERGER et al., 2015), *Feature Pyramid Network* (FPN) (LIN et al., 2017) e a PSPNet (ZHAO et al., 2017). Em adição, foi notada a popularidade dos *backbones* da família ResNet, confirmado também por (MA et al., 2019; HOESER; KUENZER, 2020; HOESER et al., 2020; NEUPANE et al., 2021). Além disso, identificou-se tecnologias como o Tensorflow (ABADI et al., 2015), PyTorch (PASZKE et al., 2019), PyTorch Lightning (FALCON, 2019), bem como bibliotecas com modelos pré-treinados como segmentation_models (YAKUBOVSKIY, 2019) e segmentation_models.pytorch (YAKUBOVSKIY, 2020).

Além disso, também foram observados aspectos como técnicas de regularização, como *weight decay* (GOODFELLOW et al., 2016), *label smoothing* (SZEGEDY et al., 2016; GOODFELLOW et al., 2016), *early stopping* (GOODFELLOW et al., 2016), *dropout* (SRIVASTAVA et al., 2014), *batch normalization* (IOFFE; SZEGEDY, 2015), *Mixup* (ZHANG et al., 2018) e *data augmentation* (GOODFELLOW et al., 2016; SHORTEN; KHOSHGOFTAAR, 2019). Com relação a esta última, foram identificadas biblitecas que foram também utilizadas no escopo da presente pesquisa: albumentations (BUSLAEV A. PARINOV; KALININ, 2018) e kornia (RIBA D. MISHKIN; BRADSKI, 2020).

Contudo, foram apontadas também as seguintes técnicas para acelerar o treinamento: *Time Based Exponential Decay* (NESTEROV, 2004), *Exponential Decay* (LI; ARORA, 2019), *Cosine Annealing* (LOSHCHILOV; HUTTER, 2017), *Cosine Power Annealing* (HUNDT et al., 2019), *One-Cycle Learning Rate Scheduling Policy* (SMITH, 2018) e *Stochastic Weight Averaging* (SWA) (IZMAILOV et al., 2018; ATHIWARATKUN et al., 2018).

Outrossim, foram constatados os seguintes conjuntos de dados (*datasets*) relevantes para a pesquisa, listando as características técnicas relevantes de cada um deles, a saber: ISPRS Potsdam e Vaihingen (ISPRS, 2012; SONG; CHOI, 2020), SpaceNet

(SPACENET, 2018; Van Etten et al., 2018), Massachusetts Buildings (MNIH, 2013), WHU building (LONG et al., 2020), INRIA aerial (MAGGIORI et al., 2017), LandCover.ai (BOGUSZEWSKI et al., 2020), AIRS (CHEN et al., 2019b) e CrowdAI (MOHANTY et al., 2020).

Além de tudo, o primeiro artigo forneceu algumas conclusões, como oportunidades de pesquisa utilizando as famílias EfficientNet (TAN; LE, 2019), ResNeSt (ZHANG et al., 2020) e SE-ResNet (HU et al., 2020), combinadas a *backbones* consagrados como U-Net, PSPNet e FPN. Além disso, identificou-se a oportunidade de testar novas arquiteturas como a HRNet-OCR (TAO et al., 2020) e a HRNetV2-OCR+PSA (LIU et al., 2021), bem como o uso de diferentes funções de perda como a exponential linear unit (ELU) (CLEVERT et al., 2016), uso de diferentes *optimizers* e o emprego no treinamento de técnicas como *Stochastic Weight Averaging* (SWA) e *Mixup*.

5 Aplicações de Segmentação Semântica

Tendo em vista todos os conhecimentos identificados na subseção 4.1, foi realizada uma seleção de métodos para testes preliminares. Nesta etapa da pesquisa, desejava-se obter a máscara segmentada no formato *raster* e, em seguida, utilizar processos de transformação de *raster* para vetor e outros métodos de pós-processamento para obter os polígonos, conforme ilustrado na figura 4.

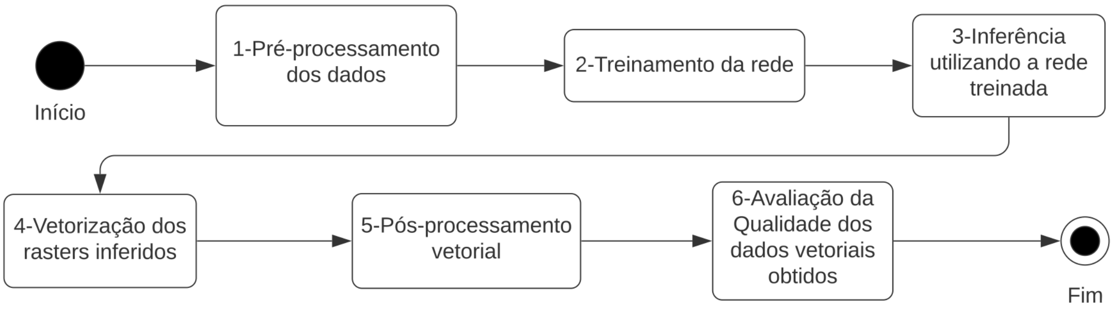

Início
1-Pré-processamento dos dados
2-Treinamento da rede
3-Inferência utilizando a rede treinada
4-Vetorização dos rasters inferidos
5-Pós-processamento vetorial
6-Avaliação da Qualidade dos dados vetoriais obtidos
Fim

Figura 4: Fluxograma de obtenção de dados planejado inicialmente para a pesquisa.

Porém, como só havia o conhecimento teórico das estruturas estudadas, desejou-se identificar o poder computacional necessário para realizar o treinamento de métodos baseados em *Deep Learning*, bem como verificar a acurácia que se conseguia obter, de forma que se pudesse avaliar o custo-benefício de arquiteturas mais complexas versus as mais simples, sob o ponto de vista de tempo de treinamento e das métricas de avaliação resultantes.

Ademais, deu-se prioridade para os *backbones* da família EfficientNet, tendo em vista que a GPU disponível para a pesquisa era apenas uma RTX-2080 Ti, com 11 Gb de memória dedicada. O uso do referido *hardware* implicava em escolhas que visavam minimizar o uso de memória de vídeo, para que se conseguisse treinar com *batches* maiores, como os *backbones* EfficientNet-B0 e EfficientNet-B1. O tamanho maior dos *batches* é de suma importância para a convergência, dada a natureza estocástica do processo de treinamento, conforme apontado por Goodfellow et al. (2016).

Entretanto, durante o desenvolvimento dos testes mencionados, o 1º Centro de Geoinformação realizou a aquisição de um servidor dedicado para *Machine Learning*, com três GPUS NVIDIA Tesla V-100, cada com 32 Gb de memória dedicada. Sendo assim, tal equipamento foi disponibilizado para a referida pesquisa, proporcionando um ganho incomum de poder computacional aos pesquisadores.

Dessa forma, tornou-se viável o treinamento de arquiteturas mais complexas e com maior número de parâmetros pudesse ser utilizada, como as combinações U-Net + SE-ResNeXt-101, FPN + ResNet-152 e FPN + SE-ResNeXt-101. Vale ressaltar que a

U-Net (RONNEBERGER et al., 2015) e a Feature Pyramid Network (FPN) (LIN et al., 2017) foram escolhidas devido à popularidade apontada no levantamento do estado da arte realizado no capítulo 4.

Já os *backbones* SE-ResNeXt-101 e ResNet-152 foram escolhidos para avaliar o custo-benefício entre a precisão dos resultados e o tempo maior de treinamento, uma vez que ambos possuem uma quantidade de parâmetros bem maior que os *backbones* da família EfficientNet. Além disso, como durante a presente etapa da pesquisa diversas publicações utilizando mecanismos de atenção foram realizadas, a estrutura SE-ResNeXt-101 foi escolhida também por conta de utilizar um mecanismo de atenção denominado *squeeze and excitation* (HU et al., 2020).

O presente capítulo é organizado da seguinte forma:

- A subseção 5.1 apresenta o artigo de título ”Building Footprint Extraction Using Deep Learning Semantic Segmentation Techniques: Experiments And Results” (BORBA et al., 2021b), apresentado no International Geoscience and Remote Sensing Symposium 2021 (IGARSS 2021) e publicado nos anais do referido evento; e

- A subseção 5.2 apresenta uma discussão complementar dos resultados apresentados em Borba et al. (2021b).

5.1 Artigo de Aplicações

BUILDING FOOTPRINT EXTRACTION USING DEEP LEARNING SEMANTIC SEGMENTATION TECHNIQUES: EXPERIMENTS AND RESULTS

Philipe Borba[1,2], *Felipe de Carvalho Diniz*[1], *Nilton Correia da Silva*[3], *Edilson de Souza Bias*[2]

[1] Brazilian Army Geographic Service (DSG)
Quartel General do Exército, Bloco F, 1º Andar, Setor Militar Urbano, Brasília-DF. ZIP Code 70630-901
[2] Geosciences Institute of the University of Brasília (UnB)
Campus Universitário Darcy Ribeiro, Brasília-DF, Brazil. ZIP Code 70919-970.
[3] University of Brasília, Campus Gama
Setor Leste (Gama), Brasília-DF, Brazil. ZIP Code 72444-240

ABSTRACT

Deep Learning Semantic Segmentation are techniques based on convolutional neural networks that have been employed in several research applications in Computer Sciences. In recent years, Remote Sensing researchers have used such techniques and achieved remarkable results. In this research paper we employ semantic segmentation techniques to extract building footprints from remote sensing imagery. We use a custom dataset called Brazilian Army Geographic Service Building Dataset to train several neural network architectures such as U-Net and FPN, combined with the following backbones: EfficientNet-B0, EfficientNet-B1, SE-ResNeXt-101 and ResNet-152. To train the mentioned structures, a framework based on Keras and Tensorflow called segmentation_models_trainer (`https://github.com/phborba/segmentation_models_trainer`) was used. Transfer Learning from ImageNet weights were used, as well as data augmentation on training images.

Index Terms— Deep Learning, Semantic Segmentation, Building Footprint Extraction, Remote Sensing, Cartography

1. INTRODUCTION

Information extraction is an essential task in Remote Sensing and has been an active research area for years [1]. Notably, for applied remote sensing such as cartography, one crucial task is to extract building geometries automatically.

With the recent advances in Computer Sciences, particularly in Artificial Intelligence (AI), several new techniques have emerged to extract information from satellite imagery. One particular set of techniques is Semantic Segmentation (SS) [2, 3, 4] and it can be used to extract building footprints from imagery [5, 6, 7, 8, 9, 10, 11].

This paper will focus only on building footprint extraction techniques by studying the existing SS methods. We will delineate our experiments parameters in section 2, showcase the results in section 3 and we will draw the conclusions and suggest further research in section 4.

2. PROPOSED SOLUTION

In this research, we will test several backbones on some of the architectures available, chosen after reviewing [12, 13]. We chose the ResNet-152 [14], EfficientNet-B0, EfficientNet-B1 [15, 16] and SE-ResNeXt-101 [17] because of their size and, to the best of our knowledge, the lack of studies in Remote Sensing journals testing them. The chosen architectures in this research that will be combined with the above mentioned backbones are: Unet [18] and FPN [19].

From the chosen backbones and architectures, we carried out some experiments with the following combinations:

1. U-Net + EfficientNet-B0
2. U-Net + EfficientNet-B1
3. U-Net + SE-ResNeXt-101
4. FPN + ResNet-152
5. FPN + SE-ResNeXt-101

The current research will use Tensorflow and Keras to implement and train models written in Python. Furthermore, the segmentation models used in this research were implemented in the python package segmentation_models [20]. To carry out the training experiments we implemented a framework called segmentation_models_trainer [21], available in Python's pip package manager [1] and on GitHub [2].

We carried out the experiments for 100 epochs, we used transfer learning [22] from ImageNet weights to achieve

[1] `https://pypi.org/project/segmentation-models-trainer/`

[2] `https://github.com/phborba/segmentation_models_trainer`

faster convergence, the used batch size was 64, and the chosen optimizer was Adam [23], with a learning rate of 0.0001 and the selected activation was the sigmoid.

The dataset used in the experiments was the Brazilian Army Geographic Service Building Dataset, built by the 1st Geoinformation Center, military organization subordinated to the Brazilian Army Geographic Service. It has more than one million and six hundred thousand building footprints extracted from airborne photogrammetric imagery of the Brazilian states of Rio Grande do Sul (spatial resolution of 35cm) and Santa Catarina (spatial resolution of 39cm). Figure 1 shows the spatial distribution of the imagery and figure 2 shows an example of the imagery and the extracted data.

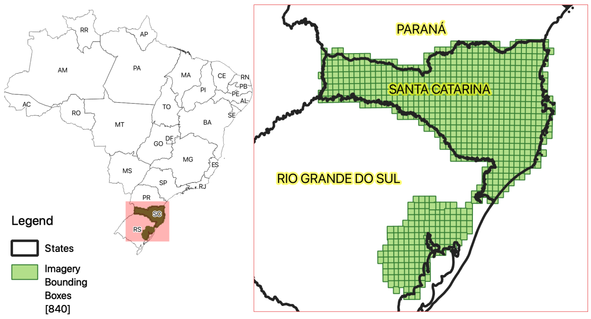


Fig. 1: Spatial distribution of the image dataset.

This dataset has not only urban scenes, but it has a large number of rural areas, one aspect that lacks in the already available datasets. We divided the dataset in the following proportion: 60% for training steps, 20% for test steps, and the remaining 20% for validation steps, all of which do not have spatial overlap.

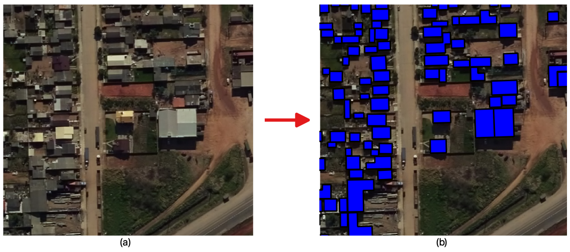


Fig. 2: Example of image (a) and extracted building footprint overlayed on the image (b)

To improve training accuracy [24], we augmented the dataset using random crop, random flips, random brightness, random contrast, random saturation and per image standardization. We built the training masks using a QGIS plugin called DeepLearningTools [25].

Furthermore, the chosen metrics for this research are mean intersection over union (mIoU), recall (R), precision (P) and F1 Score which are described by, respectively, 1, 2, 3 and 4.

$$mIoU = \frac{1}{m}\sum \frac{A_{pred} \cap A_{true}}{A_{pred} \cup A_{true}} \tag{1}$$

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

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

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{4}$$

where m is the number of expected classes, A_{pred} is the set of predictions and A_{true} is the set of Ground Truth, TP, TN, FP and FN are, respectively, the true positives, the true negatives, the false positives and the false negatives.

This research's chosen loss function is the binary cross-entropy dice loss (BCE Dice), which is the sum of the binary cross-entropy and the dice loss, which can be seen in 5.

$$\begin{aligned} L_{\text{BCE Dice}} &= -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^c \log s_i^c + 1 \\ &- \frac{2\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N}\sum_{c=1}^{C}(g_i^c)^2 + \sum_{i=1}^{N}\sum_{c=1}^{C}(s_i^c)^2} \end{aligned} \tag{5}$$

where N is the number of pixels, g_i^c is the binary indicator whether the class label c is correctly classified for pixel i and s_i^c is the corresponding predicted probability.

We carried out each training in a server with 2 Xeon Processors, each with 16 threads, 128Gb of RAM, a 2 TB SSD HD for data storage and 3 NVIDIA Tesla V-100 GPUs, each with 32 Gb of RAM.

3. RESULTS

We carried out the experiments with the parameter described in the previous section. The evaluation metrics of the experiments calculated using the validation dataset can be seen on table 1.

Architecture	Backbone	IoU	Precision	Recall	F_1
U-Net	SE-ResNeXt-101	0.451	0.518	0.774	0.621
FPN	ResNet-152	0.436	0.509	0.752	0.606
FPN	SE-ResNext-101	0.426	0.506	0.729	0.595
U-Net	EfficientNet-B1	0.413	0.489	0.728	0.588
U-Net	EfficientNet-B0	0.272	0.366	0.513	0.427

Table 1: Evaluation metrics calculated on the validation dataset of each experiment.

From the results shown on table 1, we can conclude that the combination of backbone and architecture that achieved best evaluation metrics on our experiments was the U-Net with SE-ResNeXt-101 with an IoU of 0.451, precision of 0.518, recall of 0.774 and F_1 score of 0.621. Some visual inference results can be seen on figures 3, 4 and 5.

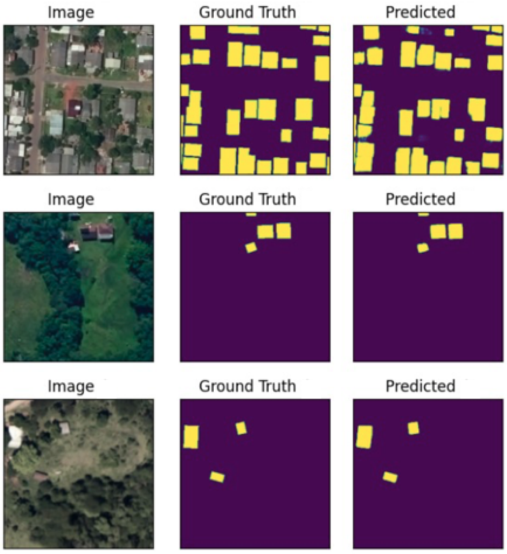


Fig. 3: Experiment results of the U-Net with SE-ResNeXt-101 backbone on urban and rural samples.

By analyzing 3, we can perceive that there is still room for improvement in the building footprints' edges. Likewise, the neural network still misses some small objects, such as the example shown in figure 4, as well as some boundary issues and small artifact that are not buildings like shown in figure 5.

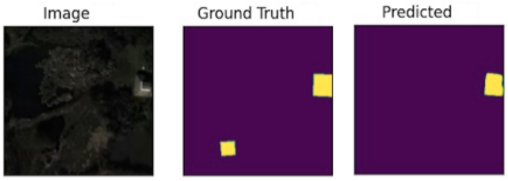


Fig. 4: Experiment results of missed predictions of the U-Net with SE-ResNeXt-101 backbone.

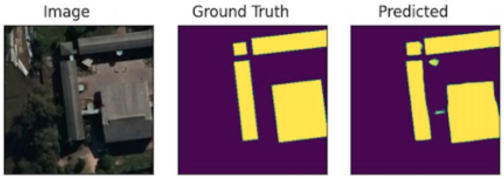


Fig. 5: Predictions with improper artifacts using the U-Net with SE-ResNeXt-101 backbone.

4. CONCLUSION

This paper has presented deep learning neural network techniques applied to Semantic Segmentation tasks to extract building footprints, trained using the Brazilian Army Geographic Service Building Dataset. It presented in section 2 the chosen architectures, backbones, technologies, dataset, evaluation metrics, loss function, optimizer, data augmentation, and hardware used to conduct experiments.

From the presented results on section 3, the best combination of architecture and backbone was the U-Net with SE-ResNeXt-101, achieving IoU of 0.451, precision of 0.518, recall of 0.774 and F_1 score of 0.621. We also concluded that there was still room for improvement in each metric by training for more epochs, since there are still building footprint edges issues and misidentified and missing small objects.

We suggest future research on other combination of architectures such as Fully Convolutional Networks (FCN) [26], PSPNet [27] and DeepLabV3+ [28]. We also suggest tests using other backbones, such as other EfficientNets [15, 16] and ResNeSt [29] backbones, as well as tests using Test Time Augmentation (TTA) and Attention Mechanisms to further improve the results and other train/test/validation splits.

5. REFERENCES

[1] Linda Shapiro, *Computer vision*, Prentice Hall, Upper Saddle River, NJ, 2001.

[2] Foivos I. Diakogiannis, François Waldner, Peter Caccetta, and Chen Wu, "ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data," 2019.

[3] Liangpei Zhang, Lefei Zhang, and Bo Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[4] Hui Yang, Penghai Wu, Xuedong Yao, Yanlan Wu, Biao Wang, and Yongyang Xu, "Building extraction in very high resolution imagery by dense-attention networks," *Remote Sensing*, vol. 10, no. 11, pp. 1–16, 2018.

[5] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L. Waslander, "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 42–55, jan 2019.

[6] Shengsheng Wang, Xiaowei Hou, and Xin Zhao, "Automatic Building Extraction from High-Resolution Aerial Imagery via Fully Convolutional Encoder-Decoder Network with Non-Local Block," *IEEE Access*, vol. 8, pp. 7313–7322, 2020.

[7] Kang Zhao, Muhammad Kamran, and Gunho Sohn, “Boundary Regularized Building Footprint Extraction From Satellite Images Using Deep Neural Network,” 2020.

[8] Wei Guo, Weihong Li, Weiguo Gong, and Jinkai Cui, “Extended Feature Pyramid Network with Adaptive Scale Training Strategy and Anchors for Object Detection in Aerial Images,” *Remote Sensing*, vol. 12, no. 5, pp. 784, mar 2020.

[9] Guang Yang, Qian Zhang, and Guixu Zhang, “EANet: Edge-Aware Network for the Extraction of Buildings from Aerial Images,” *Remote Sensing*, vol. 12, no. 13, pp. 2161, 2020.

[10] L. Hang and G. Y. Cai, “Cnn Based Detection of Building Roofs From High Resolution Satellite Images,” *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-3/W10, no. November 2019, pp. 187–192, 2020.

[11] Jingjing Ma, Linlin Wu, Xu Tang, Fang Liu, Xiangrong Zhang, and Licheng Jiao, “Building Extraction of Aerial Images by a Global and Multi-Scale Encoder-Decoder Network,” *Remote Sensing*, vol. 12, no. 15, pp. 2350, 2020.

[12] Thorsten Hoeser and Claudia Kuenzer, “Object detection and image segmentation with deep learning on earth observation data: A review-part i: Evolution and recent trends,” *Remote Sensing*, vol. 12, no. 10, 2020.

[13] Thorsten Hoeser, Felix Bachofer, and Claudia Kuenzer, “Object detection and image segmentation with deep learning on earth observation data: A review—part ii: Applications,” *Remote Sensing*, vol. 12, no. 18, 2020.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 770–778, 2016.

[15] Mingxing Tan and Quoc V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” Tech. Rep., 2019.

[16] Mingxing Tan, Ruoming Pang, and Quoc V. Le, “EfficientDet: Scalable and Efficient Object Detection,” Tech. Rep., 2019.

[17] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu, “Squeeze-and-Excitation Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vol. 9351, pp. 234–241.

[19] Tsung Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, “Feature pyramid networks for object detection,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017, vol. 2017-Janua, pp. 936–944.

[20] Pavel Yakubovskiy, “Segmentation models,” `https://github.com/qubvel/segmentation_models`, 2019.

[21] Philipe Borba, “phborba/segmentation_models_trainer: First release,” Sept. 2020.

[22] Karl Weiss, Taghi M. Khoshgoftaar, and Ding Ding Wang, *A survey of transfer learning*, vol. 3, Springer International Publishing, 2016.

[23] Diederik P Kingma and Jimmy Lei Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–15.

[24] Connor Shorten and Taghi M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *Journal of Big Data*, vol. 6, no. 1, 2019.

[25] Philipe Borba, “phborba/deeplearningtools: First release,” Oct. 2020.

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation,” Tech. Rep.

[27] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, “Pyramid scene parsing network,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 6230–6239, 2017.

[28] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam, “Rethinking Atrous Convolution for Semantic Image Segmentation,” Tech. Rep., 2017.

[29] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, Mu Li, and Alexander Smola, “ResNeSt: Split-Attention Networks,” .

5.2 Discussão do Artigo de Aplicações

A subseção 5.1 apresentou o artigo com alguns experimentos realizados à luz da pesquisa realizada no artigo apresentado na subseção 4.1. Borba et al. (2021b) realizaram experimentos com as seguintes combinações de arquiteturas e *backbones*: U-Net + EfficientNet-B0, U-Net + EfficientNet-B1, U-Net + SE-ResNeXt-101, FPN + ResNet-152, FPN + SE-ResNeXt-101.

Não obstante, para realizar os experimentos supramencionados, foi desenvolvido um *framework* baseado em Tensorflow e Keras e na biblioteca segmentation_models denominado segmentation_models_trainer (BORBA, 2020b), disponível no gerenciador de pacotes pip[1] do Python e no repositório do GitHub[2]. Durante a pesquisa realizada para o artigo em questão, também foi desenvolvido um complemento (*plugin*) para o QGIS (QGIS Development Team, 2009) denominado DeepLearningTools (BORBA, 2020a), disponível no repositório[3] de complementos do QGIS e no GitHub[4].

Além disso, também foi construído o conjunto de dados denominado *Brazilian Army Geographic Service Buildings dataset*, construído com imagens cedidas pela Secretaria de Planejamento, Orçamento e Gestão do Rio Grande do Sul (SEPLAG/RS) e pela Secretaria Executiva do Meio Ambiente de Santa Catarina (SEMA/SC), e com dados vetoriais produzidos pelo 1º Centro de Geoinformação. A autorização de uso dos referidos dados nessa pesquisa foi dada pela Diretoria de Serviço Geográfico (DSG), cujos documentos comprobatórios da solicitação de autorização para uso dos dados e o termo de compromisso do uso dos dados encontram-se respectivamente nos anexos B e C. Ao final da pesquisa, o conjunto de dados estará disponível online[5].

Outrossim, os primeiros experimentos utilizando *backbones* da família EfficientNet e os testes posteriores supramencionados permitiram que se chegasse a conclusões como: tais componentes são excelentes para situações em que o uso de memória é relevante, contudo, caso não se congele os pesos das operações de *Batch Normalization* quando o treinamento está sendo feito à partir de uma rede pré-treinada, dependendo do conjunto de dados utilizado, pode-se obter resultados não adequados ou até mesmo divergir o algoritmo.

Aliás, tal fenômeno pode explicar em partes o desempenho pior de tais estruturas no estudo em questão, visto que os experimentos que contemplavam a EfficientNet-B0 e a EfficientNet-B1 apresentaram os piores resultados em todas as métricas e não houve o cuidado de congelar os pesos das estruturas supracitadas, dado que na época

[1] https://pypi.org/project/segmentation-models-trainer/

[2] https://github.com/phborba/segmentation_models_trainer

[3] https://plugins.qgis.org/plugins/DeepLearningTools/

[4] https://github.com/phborba/DeepLearningTools

[5] https://dsgoficial.github.io/brazilian_army_geographic_service_buildings_dataset/

dos experimentos não se sabia desse fato. Outro motivo que pode explicar tais resultados é que os backbones ResNet-152 e SE-ResNeXt-101 possuem um número bem maior de parâmetros, o que propicia que a rede aprenda mais padrões nas camadas mais profundas.

Além disso, a SE-ResNeXt-101 utiliza o mecanismo de atenção *squeeze and excitation* (HU et al., 2020), que permite que o backbone aprenda mais detalhes, permitindo, dessa forma, que a rede tenha um desempenho melhor, a um custo de um uso elevado de memória. Todavia, ao se comparar os resultados das combinações SE-ResNeXt-101+U-Net e ResNet-152+FPN, percebe-se valores próximos em todas as métricas escolhidas: SE-ResNeXt-101+U-Net atingiu 0,451 de IoU, 0,518 de *Precision*, 0,774 de *Recall* e 0,621 de índice F1, enquanto a ResNet-152+FPN marcou 0,436 de IoU, 0,509 de *Precision*, 0,752 de *Recall* e 0,601 de índice F1.

Adicionalmente, o estudo em questão apontou muitas omissões, implicando em baixas métricas IoU e *Precision*. Para mais, observou-se visualmente a ocorrência de vários artefatos nas máscaras segmentadas. Além disto, nas conclusões do segundo artigo foi sugerido o uso de outras arquiteturas, como PSPNet e DeepLabV3+ (CHEN et al., 2018), de outros backbones como ResNeSt e outras EfficientNets. Por fim, também foi sugerido o emprego de outras técnicas de treinamento, como *Test Time Augmentation* (TTA), de outros mecanismos de atenção e treinamento por mais épocas.

6 Resultados

Diante dos resultados parciais até o momento da pesquisa descrito na subseção 5.2, dos objetivos inicialmente propostos para a pesquisa, das ideias apresentadas no capítulo 5 e conforme ilustrado na figura 4, a ideia era obter as máscaras de segmentação por meio de métodos semelhantes aos estudados até aquele momento, aplicar algoritmos de poligonização nos rasters, como *marching squares* (LORENSEN; CLINE, 1987), obter os polígonos e, por fim, aplicar algoritmos de pós-processamento.

Um exemplo de pós-processamento é o método para retirada de excesso do algoritmo de Douglas-Rammer-Peucker (DOUGLAS; PEUCKER, 1973; RAMER, 1972). Porém, um fato conhecido desse algoritmo é que, dependendo da escolha da tolerância, ele pode ser bem agressivo na retirada dos vértices, como apontado por Girard et al. (2021). Deseja-se encontrar um método, ou combinação de métodos que não retire vértices em excesso e que respeite o formato das edificações, conforme ilustrado na figura 5.

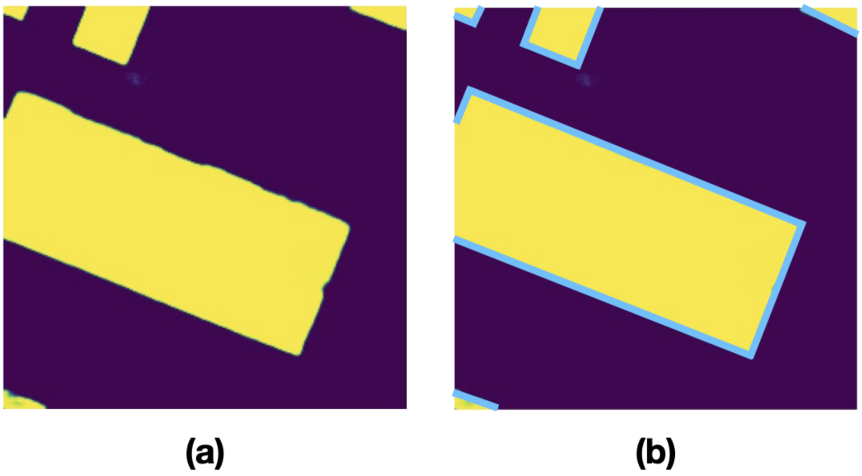


Figura 5: Exemplo de generalização dos polígonos extraídos. A imagem (a) representa a saída *raster* da rede, enquanto a imagem (b) representa, em azul, os polígonos que se deseja extrair.

Sendo assim, em sequência, durante o IGARSS 2020, os autores dessa pesquisa puderam verificar alguns métodos que abordavam o mesmo tema de extração de edificações utilizando técnicas de Deep Learning. Em particular, um trabalho que chamou a atenção foi a apresentação de título ”Regularized Building Segmentation by Frame Field Learning”(GIRARD et al., 2020).

Com a leitura de Girard et al. (2020), pode-se encontrar os códigos da pesquisa no repositório[6] do GitHub do autor. Descobriu-se também que existia um artigo de título

[6] https://github.com/Lydorn/Polygonization-by-Frame-Field-Learning

”*Polygonization by Frame Field*” (GIRARD et al., 2021), aceito na Conferência sobre Visão Computacional e Reconhecimento de Padrões 2021 (CVPR 2021), descreve com detalhes a pesquisa sobre o Frame Field. Vale salientar ainda que foram feitos vários contatos por e-mail com o Dr Girard, primeiro autor dos artigos em questão, nos quais ele de maneira muito solícita retirou várias dúvidas sobre as implementações disponíveis.

Girard et al. (2021) propõem um método denominado *Frame Field Learning*, no qual é treinada uma rede que aprende a realizar a segmentação dos polígonos, das bordas das edificações e de um campo vetorial complexo, denominado *Frame Field*, que é utilizado num método de pós-processamento, proposto pelos autores, denominado *Active Skeletonize Method* (ASM).

Tal técnica visa obter polígonos com ângulos próximos de 90 graus, segmentos sem serrilhados e visa garantir adjacências, o que é um diferencial com relação aos outros métodos estudados até então. O treinamento é feito por meio de função de perda composta, a qual é a combinação linear de 8 funções distintas que controlam diferentes aspectos do formato dos polígonos resultantes. O melhor resultado apresentado por Girard et al. (2021) foi um método em que uma ResNet-101 é combinada a uma U-Net, enquanto o *Frame Field* é acoplado ao *decoder* da U-Net.

O trabalho de Girard et al. (2021) motivou uma pesquisa detalhada sobre métodos ”*end-to-end*” para obtenção dos polígonos, ou seja, métodos que recebem imagens e extraem as estruturas desejadas no formato vetorial utilizando *Deep Learning*. Dessa busca, foi dado o início à pesquisa para o terceiro artigo.

Portanto, o capítulo atual aborda a nova pesquisa bibliográfica motivada pelos fatos supracitados, os métodos escolhidos para os experimentos e os resultados dos mesmos. O presente capítulo é organizado da seguinte maneira:

- A subseção 6.1 apresenta o artigo com os resultados da pesquisa, de título ”*Building Polygon Extraction from Very High-Resolution Remote Sensing Images using Deep Learning Methods: A meta-analysis with an experimental approach*”, a ser submetido ao *ISPRS Journal of Photogrammetry and Remote Sensing*;

- A subseção 6.2 fornece uma discussão e informações complementares dos resultados do artigo; e

- A subseção 6.3 provê uma análise complementar dos resultados do artigo da subseção 6.1, à luz das normas do SCN.

6.1 Artigo com os Resultados da Pesquisa

Building Polygon Extraction from Very High-Resolution Remote Sensing Images using Deep Learning Methods: A meta-analysis with an experimental approach

Philipe Borba[a,b,*], Edilson de Souza Bias[b], Nilton Correia da Silva[c]

[a] *Brazilian Army Geographic Service (DSG), Quartel General do Exército, Bloco F, 1º Andar, Setor Militar Urbano, Brasília-DF. ZIP Code 70630-901*
[b] *Geosciences Institute of the University of Brasília (UnB), Campus Universitário Darcy Ribeiro, Brasília-DF, Brazil. ZIP Code 70919-970*
[c] *University of Brasília, Campus Gama, Setor Leste (Gama), Brasília-DF, Brazil. ZIP Code 72444-240*

Abstract

Building footprint extraction is an important and active research field in Geosciences. This field of study covers many problems such as polygon extraction, height estimation, digital terrain model (DTM) extraction, point cloud classification, 3D building reconstruction, building detection, damage recognition, and change detection. With the recent advances in deep learning methods, many research papers that cover building footprint extraction using deep learning methods have been published in the last five years. In this paper, we perform a meta-analysis to study building footprint extraction in vector format using deep learning techniques. We identify the main research problems, most popular backbones, most used architectures, and the main methods of extracting the polygons of buildings. After the literature review, we test some techniques and propose a new method: HRNet OCR W48 Frame Field. We also propose a new building footprint extraction dataset called Brazilian Army Geographic Service (BAGS) Building Footprint Dataset.

Keywords: Meta-Analysis, Remote Sensing, Deep Learning, Semantic Segmentation, HRNet, Building Footprint Extraction, Vector Polygon Output

[*] Corresponding author
Email addresses: borba.philipe@eb.mil.br (Philipe Borba), edbias@unb.br (Edilson de Souza Bias), niltoncs@unb.br (Nilton Correia da Silva)

Preprint submitted to ISPRS Journal of Photogrammetry and Remote Sensing *January 20, 2022*

1. Introduction

DEM Extraction, 3D Modelling, Stereo Matching, and Building footprint extraction are classical problems from geosciences. Several studies from the early days of digital photogrammetry tackle these research topics, such as [1, 2, 3, 4, 5, 6, 7]. With the recent advances in the computational power of current computers and the latest development of Deep Learning (DL) techniques, many Remote Sensing (RS) research fields have greatly benefited from these computer vision breakthroughs. One particular research topic that has greatly benefited from the advances on DL is building footprint extraction.

Moreover, cutting-edge groundwork on extracting buildings from RS imagery often applies DL methods like seen on [8, 9, 10, 11, 12, 13, 14, 15]. Other examples of such techniques are studies like area monitoring using change detection methods [16, 17, 18] and post-disaster damage recognition [19, 20, 21, 22, 23].

Several recent review papers in remote sensing include assorted in-depth aspects of many applications of Deep Learning methods. [24] briefly explain what DL is, its basic building blocks, identify the most relevant journals that cover the topic, and conclude that the most popular theme from 2015-2018 is urban-related topics. [25, 26] provide a two-part, well-explained, and thorough review that provides, among other information, basic concepts of DL, the main architectures of semantic segmentation and object detection, the most popular methods, the used sensors, and the available datasets.

In addition, [27] provide a meta-analysis of semantic segmentation papers on urban environments. They cover architectures, study targets, data sources (datasets), data preparation like data augmentation and pre-processing steps, frameworks, optimizers, loss functions, evaluation metrics. Also, they identify some intriguing facts about the surveyed papers, as the appearance of salt-and-pepper noise and the insufficient training of some experiments. Besides, they point out problems like boundary pixel classification, class imbalance, and domain-shift. They also provide an annex with a table covering each paper analyzed, listing aspects like the area of each study, the dataset, the model used, framework, evaluation metric, and the highest value obtained.

Even though the previously mentioned review papers cover several trending relevant topics, they do not focus exclusively on building footprint extraction. Those that glance at this topic do not mention aspects like code availability, the usage of attention mechanisms, post-processing methods, and data output types.

In the topic of data output types, remote sensing applications, in general, often extract information as raster and perform all evaluation and data extraction in this format. However, several applications need geoinformation in vector format. For instance, vector data can be used to elaborate topographic charts [28, 29, 30], perform cartographic generalization [31], solve routing problems, perform spatial analysis and serve data in OGC web services such as WFS and WFS-T.

Likewise, GIS data can also be produced as volunteered geographic information (VGI) [32]. One example of such data production is the Open Street Maps (OSM)[1], which according to [33], has more than 5 million users from different parts of the world that collectively gather data to map the world. [33] surveys several deep learning-based techniques employed in OSM related tasks like label extraction [34], overpass extraction [35], quality control [36, 37] and vandalism detection [38].

On top of that, several private companies contribute to OSM by producing data, like Meta (formerly known as Facebook) that used weakly and semi-supervised learning to extract roads for OSM [39], and Google that extracted building footprints in the African continent [40]. However, the interest in data extraction in vector format is not restricted to private parties, since government agencies often produce geospatial data in vector format, like the USGS [41] and the Brazilian Army Geographic Service [29, 30].

Furthermore, considering the relevance of the topic of building footprint extraction and the need of studying information in vector format, the main contributions of this research are:

- We analyze the papers from the following perspectives: number of citations, input and output types, frameworks, code availability, datasets, chosen architectures and backbones, and evaluation metrics;
- We provide a focused analysis on methods that provides vector features as final outputs and compare them based on what has been published so far;
- We present a novel dataset of airborne RGB orthoimages to train building footprint extraction neural networks named Brazilian Army Geographic Service Building Dataset (BAGS Building Dataset);
- We present a new Frame Field polygonization [42] based method named HRNet OCR W48 Frame Field;

[1] https://www.openstreetmap.org/

- We conduct experiments to compare some of the surveyed vector extracting methods on the meta-analysis to the proposed method and present the results;
- We present an open-source framework called pytorch_segmentation_models_trainer [43] that implements all the proposed experiments. Their results are fully reproducible since the trained neural network weights are available on the mentioned tool's GitHub repository[2];
- We provide further conclusions based on the meta-analysis and our experiments;

We organized our work in the following manner: Section 2 presents our meta-analysis. Section 3 displays the papers explicitly related to polygon building extraction. Moreover, in section 4, we introduce the Brazilian Army Geographic Service Building Dataset and show all our experimental settings. In addition, we display the results of the experiments in section 5, and finally, in section 6, we provide the conclusions of the current study.

2. Meta-Analysis in Remote Sensing

We adopted the methodology described in figure 1 to perform our meta-analysis. We searched the Web of Science index from 2018 until 2021, with the search keywords "building extraction", filtering only by Remote Sensing publications, excluding review and conference papers, and including papers only from relevant publishers.

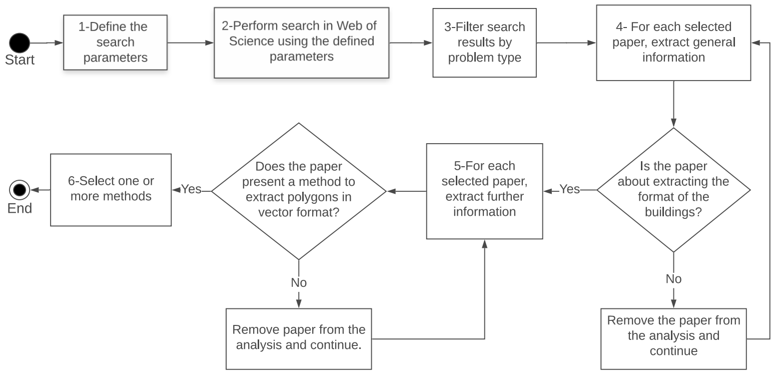

Start
1-Define the search parameters
2-Perform search in Web of Science using the defined parameters
3-Filter search results by problem type
4- For each selected paper, extract general information
Is the paper about extracting the format of the buildings?
Yes
No
Remove the paper from the analysis and continue
5-For each selected paper, extract further information
Does the paper present a method to extract polygons in vector format?
Yes
No
Remove paper from the analysis and continue.
6-Select one or more methods
End

Figure 1: General method of the meta-analysis.

[2] https://github.com/phborba/pytorch_segmentation_models_trainer

This meta-analysis section is organized as follows:

- We list the main ideas like the identified research problems and their distribution, the main techniques, the often-used datasets, the most cited papers and the code availability in subsection 2.1;
- We analyze only building footprint extraction related papers, pointing out the most used architectures and backbones, the used frameworks, and the output types in subsection 2.2;
- Finally, we analyze only building footprint extraction with vector output in subsection 2.3.

2.1. Main ideas

After applying the filters previously described, we selected 99 papers to analyze. We identified the following research problems: Building Footprint Extraction, Change Detection, Damage Recognition, Building Detection, 3D Building Reconstruction, Point Cloud Classification, DTM extraction and Height Estimation. Figure 2 shows the distribution of the papers in the mentioned research problems, where table 1 shows the list of each paper grouped by research problem.

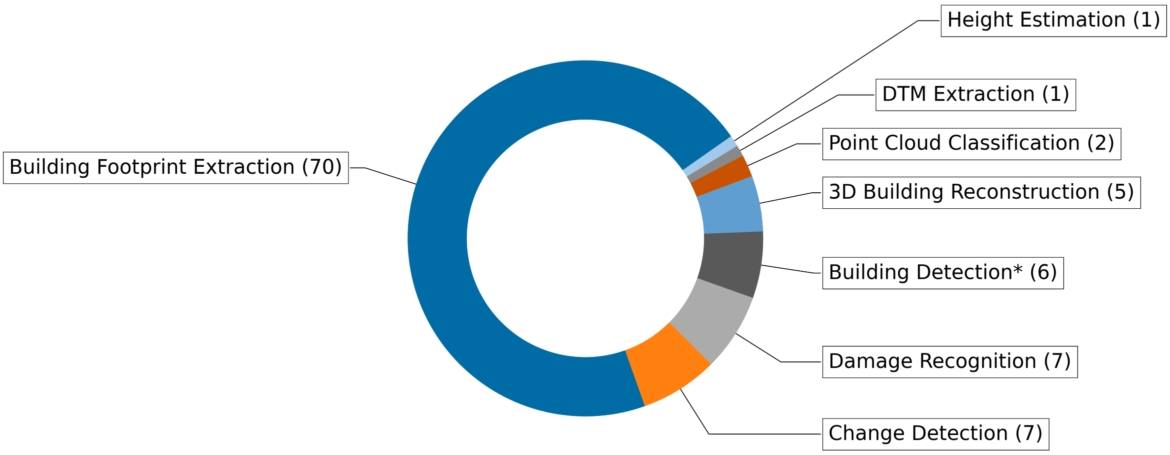


Figure 2: Problem distribution. The majority of the reviewed papers relate to Building Footprint Extraction.

Table 1: Surveyed papers grouped by problem area. The Building Detection highlighted in the table bellow considers the building object detection problems, as well as rooftop type classification.

Problem	Citation	Number of Papers
Building Footprint Extraction	[44, 45, 46, 47, 48, 49, 50, 51, 8, 52, 12, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 10, 70, 71, 72, 9, 73, 74, 75, 13, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 15, 90, 91, 92, 11, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 14, 104, 105]	70
Change Detection	[106, 107, 17, 108, 109, 18, 16]	7
Damage Recognition	[20, 110, 22, 111, 19, 112, 21]	7
Building Detection*	[113, 114, 115, 116, 117, 118]	6
3D Building Reconstruction	[119, 120, 121, 122, 123]	5
Point Cloud Classification	[124, 125]	2
DTM extraction	[126]	1
Height Estimation	[127]	1

Then, we analyzed the selected papers regarding the type of technique employed. We grouped them into 6 categories described bellow:

- Semantic Segmentation using CNNs: techniques that use only convolutional neural networks to extract feature masks. This method does not identify adjacent buildings as individual objects, and it amalgamates adjoining objects into one;

- GAN: techniques that employ Generative Adversarial Networks [128] to generate the segmentation maps;

- Super Resolution Semantic Segmentation: techniques that employ super-resolution to either improve the results of the segmentation or as a pre-processing step prior to segmentation;

- Instance Segmentation using CNNs: similar to Semantic Segmentation, this technique is based on convolutional neural networks. This method is often combined with object detection methods to individualize the pixels of each building before segmenting. This method has a disadvantage: it is possible to generate gaps and overlaps in adjacent features when the raster data is transformed to vector since the method extracts each object individually;

- LSTM: techniques that use Long Short Term Memory (LSTM), a structure often used in Recursive Neural Networks (RNNs);
- Semantic Segmentation using CNNs and LSTM: techniques that combine semantic segmentation and LSTM to extract the geometry of buildings.

Table 2 groups the surveyed papers into the categories in question and reveals that the most popular technique in this meta-analysis is Semantic Segmentation using CNNs.

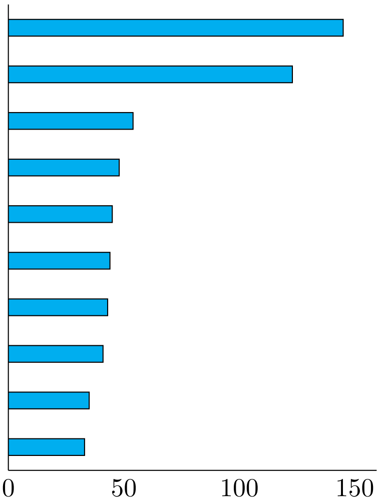
Table 2: Building footprint extraction papers grouped by type of technique.

Technique	Papers	Number of Papers
Semantic Segmentation using CNNs	[11, 44, 46, 86, 90, 69, 103, 64, 50, 8, 57, 53, 58, 60, 56, 54, 66, 91, 15, 68, 85, 84, 70, 96, 89, 48, 45, 14, 61, 95, 76, 79, 10, 83, 92, 75, 98, 87, 59, 105, 78, 12, 63, 62, 49, 104, 88, 72, 9, 13, 71, 100, 47, 80, 77, 51, 99, 81]	58
GAN	[102, 65, 94, 74, 52]	5
Super Resolution Semantic Segmentation	[82, 67, 55]	3
Instance Segmentation using CNNs	[73, 93]	2
LSTM	[101]	1
Semantic Segmentation using CNNs and LSTM	[97]	1

Next, we surveyed the ten most cited papers. From the top 10, 9 are Building Footprint Extraction problems. Only [76] belongs to the category Damage Recognition. This fact, combined with the problem with most papers shown in figure 2, shows that Building Footprint Extraction is the trending topic in the surveyed papers.

Table 3: 10 most cited papers of the search Building Extraction, from 2018-2021. In the top 10, only [76] proposes a different problem, Damage Recognition. All other 9 cover Building Footprint Extraction.

Reference	Number of Citations
[88]	145
[8]	123
[64]	54
[9]	48
[10]	45
[11]	44
[69]	43
[76]	41
[20]	35
[12]	33



Furthermore, we analyzed the chosen frameworks of each paper. The most popular framework is PyTorch [129], followed by Tensorflow and Keras with Tensorflow [130]. We considered categories with Tensorflow, other with Keras with Tensorflow, and another with only Keras because the first versions of Tensorflow did not integrate with Keras. In the early days, Tensorflow and Keras were different frameworks. From this analysis, we can perceive that 22 papers do not provide information regarding which framework was used. Figure 3 shows the framework distribution. Table 16 in annex A describes which papers use each listed framework.

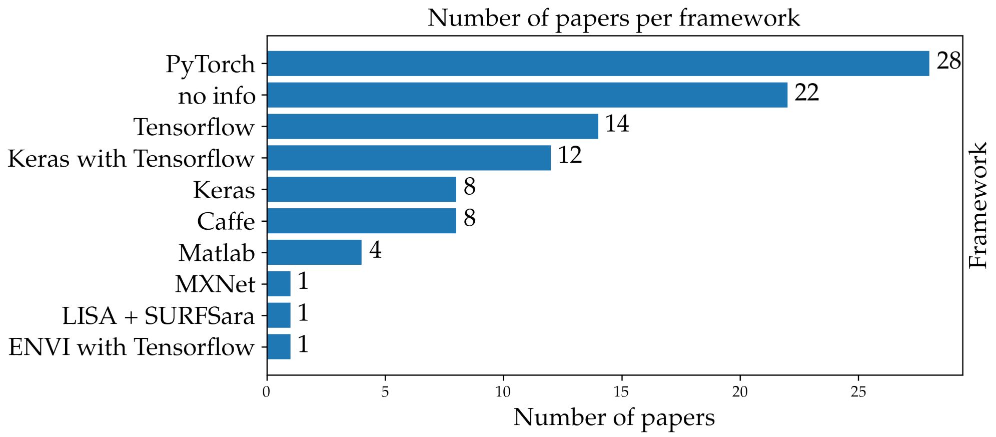


Figure 3: Frameworks.

The final analysis considering the 99 papers concerns the code availability. We considered the paper's code available if it provided a link to an external source with the code to replicate the experiment. This analysis excluded papers that only provided helper codes like pre-processing routines but not the neural network models. We also excluded papers that provided repository links with no code. Of all surveyed papers, only 13 have code available online, which points to the possibility that not all the authors of the considered researches may be adept at open science, and some of them may not be on board with producing reproducible research. Figure 4 shows this result and the links for each available code can be seen in table 17 in the annex.

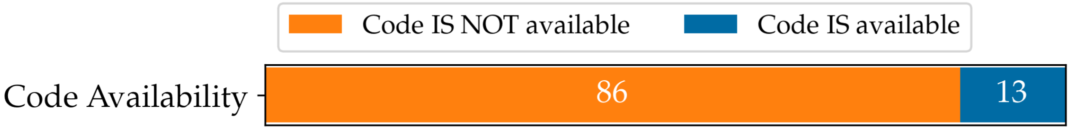


Figure 4: Number of papers with code available. We considered the paper's code available if it provided a link to an external source with the code to replicate the experiment. We excluded from this count papers that only provided helper codes like pre-processing routines, but not the neural network models. We also excluded papers that provided repository links with no code.

2.2. Building Footprint Extraction Papers Analysis

Next, we considered only the 70 building footprint papers listed in table 1. We analyzed each research and identified which datasets they chose for their study and presented the results in figure 5. It is worth mentioning that the surveyed papers can choose more than one dataset in their methodology, so the sum of occurrences in figure 5 does not add to the total of considered papers.

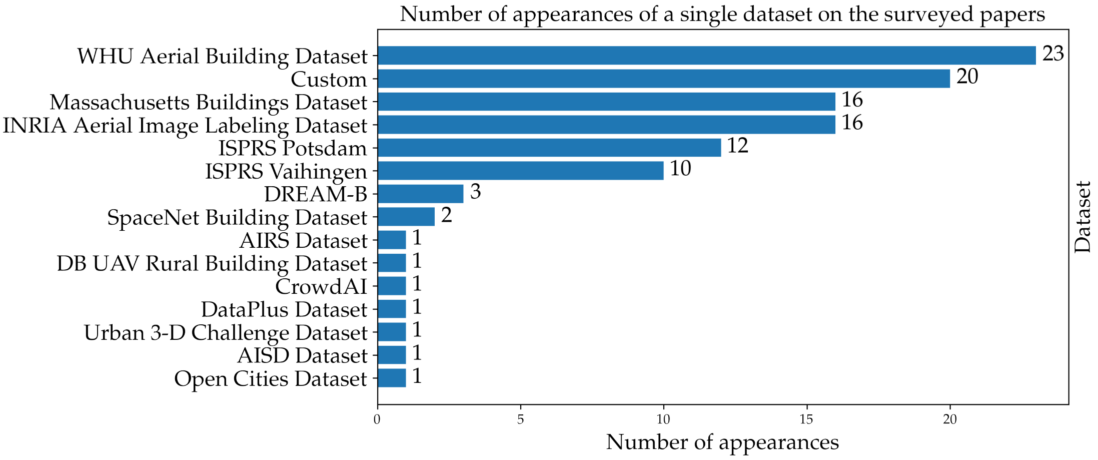


Figure 5: Datasets.

Figure 5 shows that the most popular dataset is WHU Aerial Building Dataset [131], followed by several custom datasets that we grouped into one single category, named Custom, then by Massachusetts Buildings dataset [132], INRIA Aerial Labeling dataset [133], ISPRS Potsdam and Vaihingen [134, 135]. Except for WHU Aerial Building Dataset, all mentioned only provide raster masks, making research with vector output more difficult. Examples of datasets that provide the vector as well are SpaceNet Building Dataset [136, 137], CrowdAI [138], and Open Cities Dataset [139]. Table 15 in annex A details each dataset and the papers that use them. For more information about the specs of each dataset, we recommend [27], which built a thorough list of datasets and their specs.

In sequence, we analyzed the types of inputs used in these papers. Figure 6 shows that the majority of papers consider only RGB inputs. It is worth to point out that LiDAR data is used in [64, 70, 100, 96], while near-infrared data in [64, 86, 101], Digital Surface Model (DSM) and/or normalized Digital Surface Model (nDSM) in [88, 76, 126, 123, 122, 86, 72, 87, 62], and finally

Normalized Difference Vegetation Index (NDVI) in [88, 62]. This result shows that not many building footprint extraction researches use imagery different from RGB.

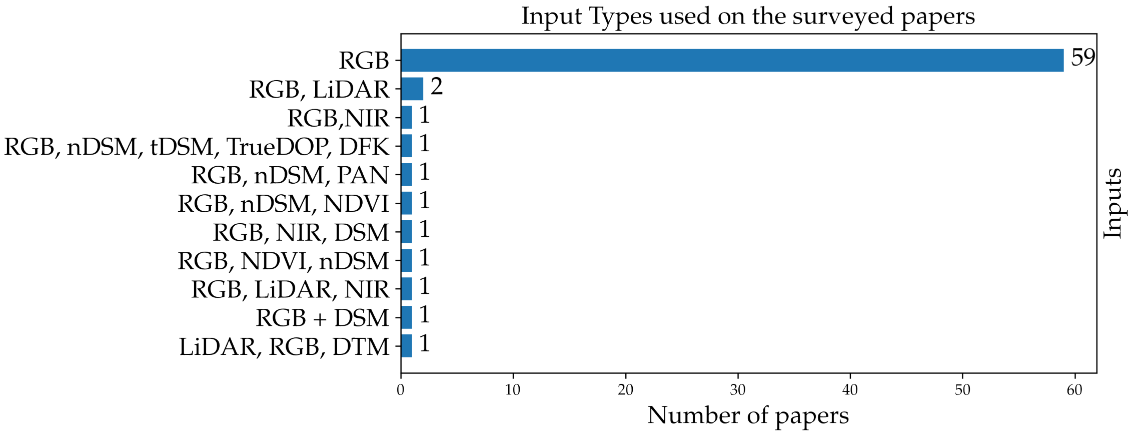


Figure 6: Building footprint extraction papers grouped by input types.

With respect to the models, we analyzed which backbones and architectures the surveyed papers chose. Table 4 groups the architectures that were combined more than once to some backbone. The result of the analysis shows that the most popular architecture is the U-Net [140], followed by CNN based and custom architectures. Other architectures worth commenting are the Feature Pyramid Network (FPN) [141], SegNet [142] and DeepLabV3+ [143]. Moreover, table 4 also show the popularity of ResNet based backbones, as well as some variations like Xception [144], InceptionV3 [145] and ResNeXt-50 [146]. The complete analysis, where we listed which papers use each architecture group and backbone is available in table 18 in the annex.

Table 4: Architecture groups associated more than once to some particular backbone. The detailed version of this table can be found in table 18 in the annex.

Architecture Group	Backbone	Number of Papers
U-Net based	VGG, Custom, Siamese, Custom, no info, Xception, ResNet-34, NASNet-Mobile, Custom (ESPC module), NASNet, classical from original papers, VGG-16, ResNet	24
CNN based	Custom, RetinaNet, InceptionV3, MobileNet, ResNet-101, classical from original papers	6
Custom	Custom, VGG-19 and SegNet, Modified DarkNet-53, VGG-16, ResNet-50	6
FPN based	ResNet-101, Custom, ResNet-50, ResNeXt-50	5
FCN based	Custom, VGG-19, Custom VGG-16	4
SegNet	Custom, VGG-16	3
MAP-Net	Custom, MAP-Net based, Custom	2
DeepLabV3+	ResNet-101, Xception	2
DenseNet	Custom	2

Finally, we analyzed the output types of the selected papers. From the 70 papers, 63 deal only with raster segmentation, two deal with both raster and vector, and 5 study vector output. Figure 7 shows the mentioned results. From this analysis, we selected these seven papers and analyzed them in depth in section 2.3.

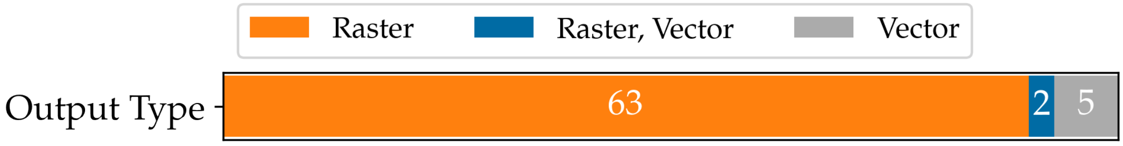


Figure 7: Building footprint extraction papers grouped by output types.

2.3. Building Footprint Extraction Papers With Vector Output Analysis

After analyzing the distribution of output types in the building footprint extraction papers in section 2.2, we selected them for further investigation. Table 5 shows the selected papers details.

Table 5: Building footprint extraction papers with vector output.

Paper	Input Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[69]	RGB	SpaceNet	U-Net	classical from original papers	Binary Cross Entropy Loss	random rotation, random scale	IoU, Precision, Recall, F1 Score
[84]	RGB	INRIA, Massachusetts Buildings	MFCNN (U-Net based, with Pyramid Aggregation Unit)	Custom	Joint loss (cross entropy loss, dice loss)	random crop, random rotation, random noise	Overall Accuracy, Precision, Recall, F1 Score, IoU
[78]	RGB	AIRS	PSPNet	ResNet-50	Cross Entropy Loss, Bi-projection loss, Relative shape loss	no info	IoU, VertexF, VertexP, VertexR
[96]	RGB, LiDAR	Custom	U-Net	no info	Weighted Binary Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score, IoU
[86]	RGB, NIR, DSM	ISPRS Potsdam	SegNet	VGG-16	no info	random scale, random rotation, random flip	Overall Accuracy, Precision, Recall, F1 Score, IoU, mIoU
[97]	RGB	CrowdAI, Open Cities	FPN	ResNet-101	Cross Entropy Loss, Smooth L1 Loss	no info	PoLiS, IoU, Average Precision
[103]	RGB	Custom	DeepLabV3+	ResNet-101	Joint loss (cross entropy loss, dice loss)	random flip, random color	Precision, Recall, F1 Score, IoU

Then, from the analysis of 5, we draw an immediate conclusion: most papers used evaluation metrics like IoU, Precision, Recall, and F1-score. We analyzed each individually, and all evaluated the raster output, except [78], which evaluated the IoU in the vector output and also used metrics like Vertex F1-score (VertexF), Vertex Precision (VertexP), and Vertex Recall (VertexR). Another exception is [97], which computes IoU and Average Precision by rasterizing their vector output. They also compute a different metric, called Polygon and Line Segments (PoLiS), defined by [147].

Thus, another immediately perceived aspect is that most use a similar set of data augmentation techniques, like random rotation, random scale, and random flip. In particular, [103] also use random color data augmentation, [84] use random noise in their training, and [78, 96, 97] provide no info on the usage of the technique in question.

Moreover, from the selected papers, only [96, 86] use inputs other than RGB, like LiDAR, NIR, and DSM. For instance, [86] used RGB, NIR, and DSM data from an unmanned aerial vehicle (UAV) to train a SegNet with a VGG-16 backbone. They also compared their proposed method to ISPRS Potsdam data. They did not inform which loss function they chose, but they described using data augmentation. They obtained their best results when using RGB and DSM data as training inputs.

Furthermore, [96] use RGB and LiDAR data to train a U-Net with their custom dataset, built with imagery from a neighborhood of Warsaw, the Polish capital and data downloaded from the geoportal[3] of the Head Office of Geodesy and Cartography, the Polish National Mapping Agency. Their images have a spatial resolution of 0.1 m, and their LiDAR data have a density of 12 points per m^2. They trained their neural network using the weighted binary cross-entropy loss, and they did not tell whether they used data augmentation or mention which backbone they used. They processed the output raster masks to extract polygon contours, which were post-processed using a Douglas-Peucker simplification algorithm [148] to remove extra vertexes, but they do not say which parameter they used in this post-processing step. They discussed their results on the raster segmentation data, showing their overall accuracy, precision, recall, F1-score, and Per-object IoU mean. [96] also compared the average shift of the output post-processed polygons to the ground truth, but they did not tell how they measured this shift: average vertex shift or geometry displacement.

Like [96], only [84, 103] also proposed post-processing methods to extract their polygons. [84] proposed the usage of Morphological filtering algorithms for building outline optimization. [103] suggested the use of Morphological opening prior to polygonization using contours. Then, they used Douglas-Peucker simplification, pixel to geographic coordinate transform, polygon merging,

[3] https://mapy.geoportal.gov.pl/imap/Imgp_2.html

and, finally, area increase to compensate for the effects of the morphological opening.

Besides, all other papers [69, 84, 78, 97, 103] used only RGB as their input data. Among these papers, it is worth mentioning some of their choices on architecture and backbones. [69, 84] trained U-Net based networks, where [84] proposed a custom U-Net, with a Pyramid Aggregation Unit called MFCNN. [78] proposed using a PSPNet with a ResNet-50 backbone, combined with a PointCNN branch to extract polygons. Additionally, [97] proposed a method that uses an FPN object detection network with a ResNet-101 as the backbone and a PolygonRNN coupled with the feature map output of the object detection localization branch. [103] trained a DeepLabV3+ with a ResNet-101 backbone.

In addition, all papers that used only RGB used some cross-entropy loss, either as their only loss, like [69], or as a part of a joint loss, like cross-entropy loss combined with the dice loss [84, 103]. In particular, [78] enforced in the modified PointNet branch of their method the cross-entropy loss combined with a bi-projection loss and a relative shape loss.

From this analysis, methods that attracted our attention were [78, 97], since they propose end-to-end methods that output polygons. Also, [84, 103] caught our eyes because of their post-processing methods. However, we want to select methods that do not solemnly rely on Douglas-Peucker simplification because this algorithm can be very vertex greedy and, depending on the chosen tolerance, can cause deformities in the post-processed data.

3. Related Work with Polygon Output

In section 2, we surveyed papers from Remote Sensing journals, analyzed 99 papers, then narrowed it down to 70 building footprint extraction studies. Then, from the 70 papers, we restricted the search to 7 papers that extracted buildings in vector format. Moreover, in section 2.3, we selected two methods: [78, 97], which presented end-to-end deep learning-based methods. In this section, we will dive deeper into the details of these techniques, and we will conduct a brief review in Computer Sciences papers that output polygons in vector format to find out more methods that should be tested in the experimental section of this paper.

As we briefly explained in section 2.3, [78] proposes a method called PolygonCNN, a neural network with the PSPNet architecture combined with a ResNet-101 backbone. It also combines a modified PointNet, originally used in point cloud deep learning-based studies. This component receives the vertexes of the contours extracted from the PSPNet feature map binarized output and tries to output the coordinate list of the built polygon.

In addition, to train their neural network, [78] propose two losses that measure the similarity of polygons: the bi-projection loss and the relative shape loss. They compare their proposed losses with a loss based on the Chamfer Distance (CD) [149], like used by [150, 151], as pointed out by [78] in their paper. They compare their methods to the PolygonRNN [152], DARNet [153] and show that their proposal outperforms the others in IoU, VertexF, and VertexR. Also, they test the usage of the three losses previously mentioned and show that they achieved their best results when using their novel shape loss.

However, this method expects a single polygon in the image input of the neural network, which implies that the analyst using this method should crop the images so that each has one considered polygon. Thus, this is a disadvantage in practical applications since the GIS analysts want to input several images and extract the polygons without localizing them prior to the method.

Thus, the localization need described above is also found in some computer sciences methods, like PolygonRNN [152], PolygonRNN++ [154], DARNet [153], Curve-GCN [155], and Conv-MPN [156]. PolygonRNN, PolygonRNN++, Curve-GCN, and Conv-MPN have the localization problem previously cited because they were conceived as part of semi-automatic annotation frameworks, which has as prior the user selecting the bounding box of the object to be annotated in the image.

Additionally, PolygonRNN and PolygonRNN++ are Recursive Neural Network (RNN) based methods, which receive a cropped image with a single object that is being segmented and outputs a sequence that is converted to a list of pixel coordinates, which represent the segmented polygon. This method can produce good results but is very memory-consuming, according to [78, 42, 97].

Similarly, Deep Active Ray Network (DARNet) is a model based on active contours and CNNs. According to [153], their CNN predicts the feature maps, and then they compute the energy landscape to apply the active contours method iteratively. [153] point out that their method has deficiencies when dealing with adjacent structures such as adjacent buildings.

Besides, Curve-GCN is a graph convolutional network (GCN) [157] based method that uses a multi-layer GCN called Graph-ResNet, like [158, 159]. Since it processes one object at a time, it does not consider neighbor objects or any topological structure. So, as the method does not consider adjacencies, adjacent objects can have overlapping annotations.

Therefore, Conv-MPN is a Message Passing Neural [160] (MPN) based network named Convolutional Message Passing Network (Conv-MPN). It relies on a graph convolutional network [157] to compute the vertexes of the polygon from the output of a Dilated Residual Network. According to the authors, this method is memory-consuming and has some detection issues since some of the polygons cannot be closed by Conv-MPN. Thus, this method is unsuitable for this research since building footprint extraction methods must produce closed polygons.

Since all methods previously described rely on prior localization, we carried out further research to find techniques that already performed the object localization. As a result of this search, we found a method influenced by PolygonRNN called PolyMapper proposed by [161]. PolyMapper [161] solves the deficiencies of PolygonRNN by coupling the PolygonRNN input to the output of the RoiAlign operation of an object detection network, solving the localization problem described at the beginning of this section. They use a VGG-16 as the backbone of the object detection module, combined with an FPN. In their paper, [161] test their proposed technique in building footprint extraction and road extraction problems. In particular, they outperform Mask R-CNN [162] and PANet [163] methods on CrowdAI dataset.

[97] proposed a method that is a variation of PolyMapper. In the context of this research, we will call the method proposed by [97] ModPolyMapper. Instead of a VGG-16 as the backbone, ModPolyMapper the ResNet-101. [97] trained their neural network proposal on Open Cities Dataset and CrowdAI and claimed that their method outperforms the original PolyMapper, Deep snake [164], and Mask R-CNN [162]. In their ablation studies, they also propose using attention mechanisms like Global Context Blocks (GCB) [165], an attention block that is the combination of the Channel Attention Block (CAB) [166] and the Spatial Attention Block (SAB) [167], and one of their proposals: the Boundary Refinement Block (BRB). They show that these attention mechanisms can improve the results of ModPolyMapper at marginal gains, which in our opinion may not justify the memory increase trade-off. Since this method is based on an object detection model, each polygon is processed independently, leading to occasional gaps and overlaps in the extracted polygons.

Our search for alternative methods also led us to polygonization by Frame Field, proposed by [42]. In this method, the authors propose the usage of a U-Net with a Resnet-101 backbone and a branch coupled to the feature map output of the U-Net, a structure that outputs a four-dimensional feature map that represents a complex field named Frame Field. In addition, they propose a polygon extracting technique: the Active Skeletons Method (ASM). ASM uses the Frame Field to find the building corners and keep them in the Douglas-Peucker (DP) simplification, mitigating the vertex greedy nature of DP. This method extracts the whole input tile's polygons and respects the adjacencies between neighbor polygons.

4. Methodology

We carried out several experiments using some of the techniques shown in sections 2 and 3. After defining the methods, we also chose the training and test datasets, the hyperparameters, and the evaluation metrics. We describe the methodology adopted for each experiment in the flowchart shown in figure 8.

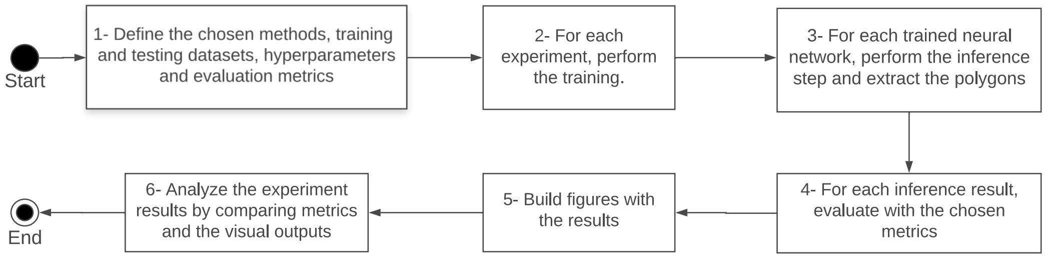

Start
1- Define the chosen methods, training and testing datasets, hyperparameters and evaluation metrics
2- For each experiment, perform the training.
3- For each trained neural network, perform the inference step and extract the polygons
4- For each inference result, evaluate with the chosen metrics
5- Build figures with the results
6- Analyze the experiment results by comparing metrics and the visual outputs
End

Figure 8: Experiment flowchart.

This section is organized as follows: subsection 4.1 details the methods, subsection 4.2 describe the chosen datasets, subsection 4.3 list all training techniques, hardware and software employed in each test and, subsection 4.4 informs the chosen evaluation metrics.

4.1. Methods

As shown in section 3, there are several options that we could choose for our research. PolygonRNN [152], PolygonRNN++ [154], DARNet [153], Curve-GCN [155], Conv-MPN [156], and PolygonCNN [78] are semi-supervised methods that need previously cropped images. We want to test trully end-to-end methods, i.e. methods that have RGB images as inputs and outputs polygons.

Moreover, we chose ModPolyMapper [97] and Polygonization by Frame Field [42] as our methods because they are end-to-end. They claim that their approach outperforms the current SOTA methods like PolyMapper. These two methods were proposed in a similar time range. From our bibliographic survey, we did not find papers comparing these techniques, so we identified that comparing these procedures is an excellent research opportunity.

18

ModPolyMapper uses a ResNet-101 combined with an FPN as the backbone of a Mask R-CNN [162] object detection network. The output of the RoiAlign operations is combined with the RNN component, similar to PolygonRNN. The losses used in the training of this method are binary cross-entropy for the RNN part, the object detection losses, the classification loss, the RoIAlign loss, and the RPN loss. Figure 9 shows a schematic representation of the ModPolyMapper. We did not use the proposed attention mechanisms described in the figure, whose results were shown in their ablation studies. We decided that the marginal gains presented by [97] when using such mechanisms did not justify the memory trade-off caused by them.

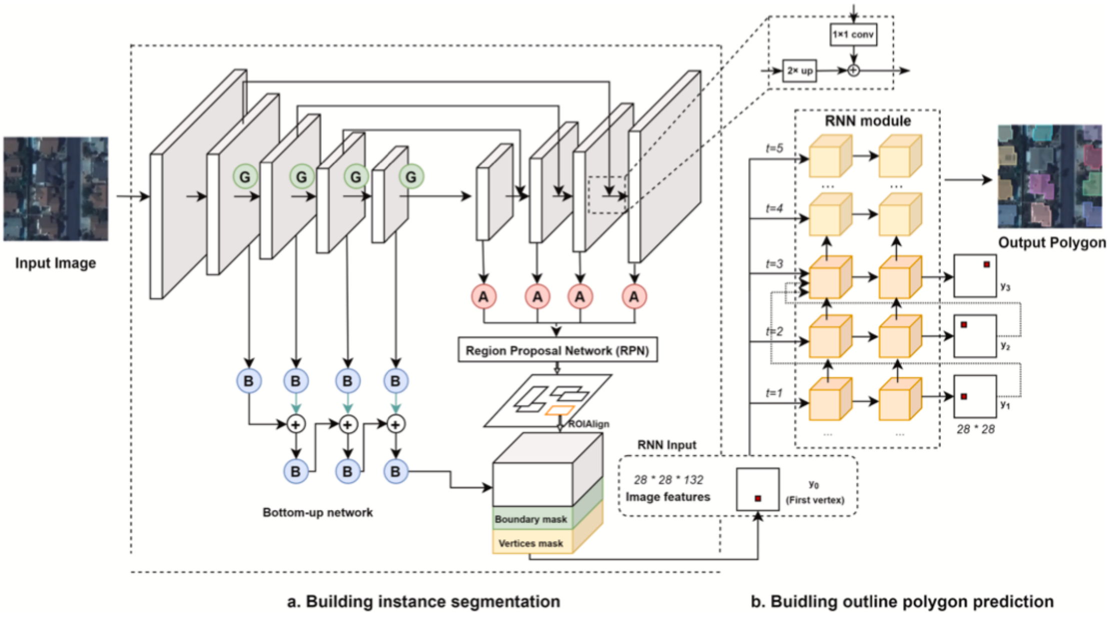


Figure 9: Description of the ModPolyMapper. Extracted from [97].

In addition, [42] claims that their best result on the AICrowd dataset is the U-Net with a ResNet-101 as the backbone. Their method proposes a composition of losses, as shown in figure 10. We used the same loss coefficients as suggested in the annex of [42].

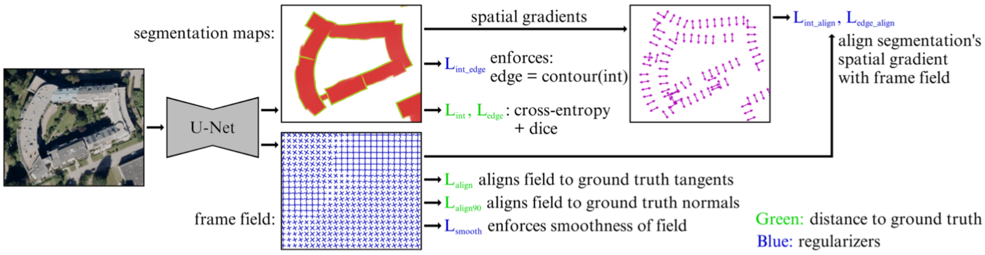

segmentation maps:
U-Net
frame field:
spatial gradients
L_{int_edge} enforces: edge = contour(int)
L_{int}, L_{edge}: cross-entropy + dice
L_{int_align}, L_{edge_align}
align segmentation's spatial gradient with frame field
L_{align} aligns field to ground truth tangents
$L_{align90}$ aligns field to ground truth normals
L_{smooth} enforces smoothness of field
Green: distance to ground truth
Blue: regularizers

Figure 10: Frame Field method description. Extracted from [42].

In our experiments, we tested the best method proposed by [42]. We also wanted to study different architectures coupled to the idea of the Frame Field. So, we proposed two variations: one using the HRNet OCR W48, which was proposed by [168, 169], and the other using DeepLabV3+, proposed by [143]. We chose the HRNet since it is currently the best method reported on the platform Papers With Code [170] on the Cityscapes test and Cityscapes val dataset competitions, reported as HRNetV2+OCR+, while the DeepLabV3+ was used in some of the reviewed papers, as well as it held the top results in 2017 in the Cityscapes competition. Figure 11 shows the architecture of HRNet, figure 12 shows the HRNet combined with the Object Context Module (OCR) and figure 13 shows the structure of the DeepLabV3+.

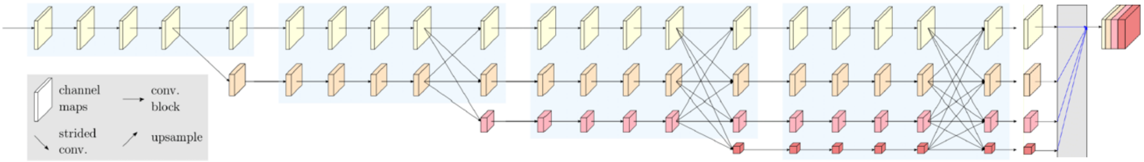


Figure 11: HRNet schema. Extracted from [171]

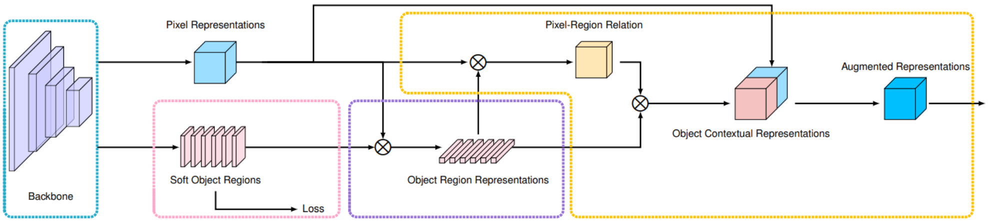

Pixel Representations
Pixel-Region Relation
Augmented Representations
Object Contextual Representations
Backbone
Soft Object Regions
Loss
Object Region Representations

Figure 12: OCR schema. Extracted from [171]

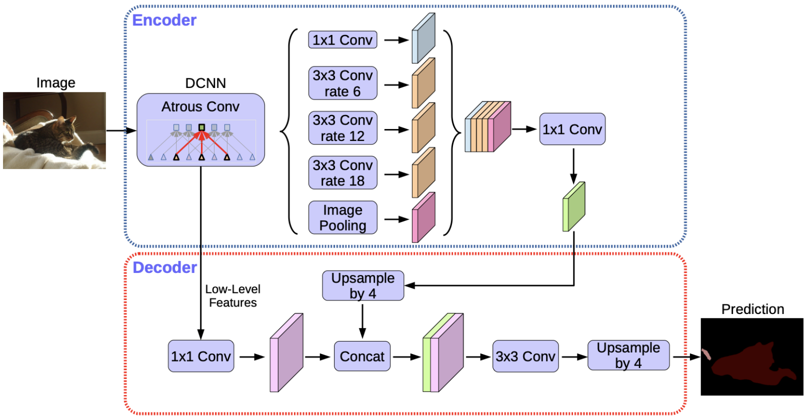


Figure 13: DeepLabV3+ description. Extracted from [143].

Since each backbone has a specific size, we trained the chosen methods respecting these input sizes. Figure 6 shows the list of the input sizes according to the backbones.

Table 6: Input sizes of the backbones used in the experiments.

Backbone	Size (pixels)
ResNet-101	224 × 224
DeepLabV3+	256 × 256
HRNet OCR W48	300 × 300

We proceeded in the inference procedures of ModPolyMapper with a threshold of 85% on the detected bounding boxes, used the sequence length of 60 and grid size of 28 in the PolygonRNN head. Regarding the inference procedures of Fame Field-related methods, we padded the input image with mirroring padding so that the output format would be a multiple of the input shape. Then, we tiled the padded image, proceeded with the network inference, integrated the tiled inferences, and performed a center crop to restore the original image size. The coefficients of the polygonization loss parameters of the ASM can be shown in table 7, the angle parameters in table 8, and the other parameters used in the polygonization in table 9.

Table 7: Active Skeletons Method polygonization parameters.

	step_thresholds	data	crossfield	length	curvature	corner	junction
Values	0	1.0	0.0	0.1	0.0	0.0	0.0
	100	0.1	0.05	0.01	0.0	0.0	0.0
	200	0.0	0.0	0.0	1.0	0.5	0.5
	300	0.0	0.0	0.0	0.0	0.0	0.0

Table 8: Active Skeletons Method angle parameters.

	corner_angles	junction_angles	junction_angle_weights
Values	45°	0°	1.00
	90°	45°	0.01
	135°	90°	0.10
		135°	0.01

Table 9: Active Skeletons Method other parameters.

Parameter	Value
init_method	skeleton
data_level	0.500
curvature_dissimilarity_threshold	2.000
junction_angle_threshold	22.500
learning_rate	0.001
gamma	0.995
tolerance	1.000
seg_theshold	0.500
min_area	10.000

4.2. Datasets

In section 2, we showed some building footprint extraction datasets such as: INRIA [133], AIRS [172] and ISPRS Potsdam [134, 135]. These datasets provide the pre-built raster masks and do not provide the polygons in vector format that originated the masks. So, in the context of this study, they are not suitable.

We carried out our experiments using two datasets: the Brazilian Army Geographic Service (BAGS) Building Dataset, built in the context of this research and detailed in section 4.2.1, and the AICrowd, detailed in section 4.2.2. We chose this last dataset to compare our experiments with the original papers' results [97, 42], described in subsection 4.1.

4.2.1. Brazilian Army Geographic Service Building Dataset

The Brazilian Army Geographic Service (BAGS) Building Dataset[4] is an open dataset that has been briefly described by [173]. The 1st Geoinformation Center (a branch of the Brazilian Army Geographic Service) built this dataset with airborne RGB photogrammetric images from SEPLAG/RS (spatial resolution of 35 cm) and SEMA/SC (spatial resolution of 39cm) that respectively cover portions of the Brazilian States of Rio Grande do Sul and Santa Catarina. Figure 14 shows the imagery coverage of the images from SEPLAG/RS and SEMA/SC.

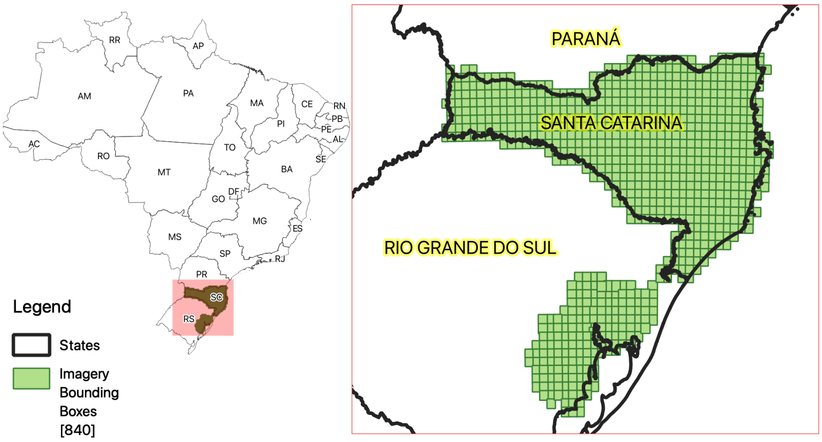


Figure 14: Spatial distribution of the tiles of the BAGS Buildings dataset. Extracted from [173].

[4] https://dsgoficial.github.io/brazilian_army_geographic_service_buildings_dataset/

In addition, the 1st Geoinformation Center extracted more than 1 million and six hundred thousand building footprints from the imagery mentioned above. With pytorch_segmentation_models_trainer, we built the following mask images used in the paper's experiments: boundary, interior, and distance masks for the Frame Field Method; bounding boxes and polygon embeddings for ModPolyMapper. The total amount of image tiles of the dataset is 247,713, ranging from $512px \times 512px$ to $573px \times 573px$. The chosen dataset split was 80% for training steps and 20% for test steps. We used the QGIS plugin DeepLearningTools[5] [174] to visualize the built tiles and manually fix some errors that might have occurred. Figure 15 shows some examples of the dataset tiles.

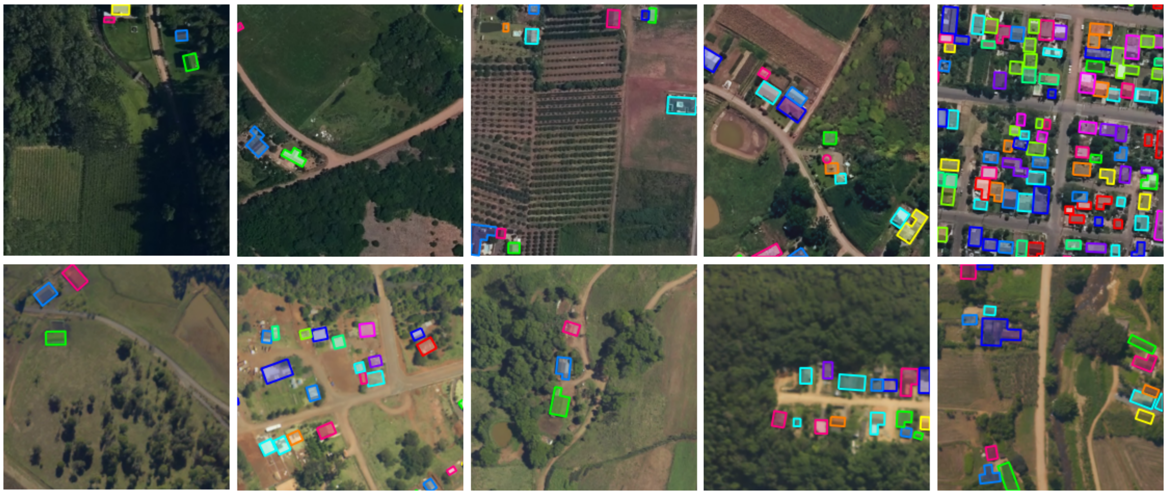


Figure 15: BAGS Building dataset train tiles examples.

We built this dataset as an alternative to the existing ones because the already available data often only portrait urban environments from the USA and Europe. This dataset offers more different cases of rural landscapes of South America, with high-resolution imagery and a significant amount of extracted building footprints in vector format.

4.2.2. AICrowd

The AICrowd [138] is a dataset built using only SpaceNet [136, 137] RGB images. It has 280,741 training images, 60,317 validation images and 60,697 test images, and each tile has 300 x 3000 px.

[5] `https://plugins.qgis.org/plugins/DeepLearningTools/`

Figure 16 shows some examples of training tiles with the polygons overlayed.

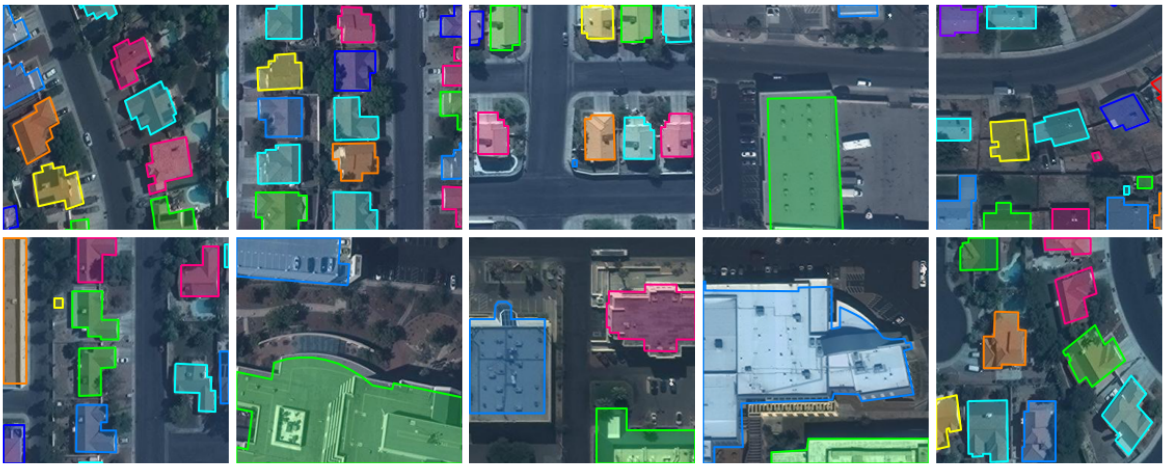


Figure 16: AICrowd train tiles examples.

4.3. Experimental Setup

In our experiments, we used AdamW [175] as our optimizer. To avoid overfitting, we used random crop, random flips, and histogram jitter as data augmentation techniques, and we also enforced a 10^{-3} weight decay (also known as L_2 regularization) factor in our experiments.

Furthermore, we also used one cycle learning rate scheduler [176] to converge faster, as shown by [177]. We employed gradient clipping to avoid exploding and vanishing gradients [178, 179], which often occur on RNN based methods. We also used gradient clipping for quick convergence [180].

Moreover, we employed stochastic weight averaging [181] in the last 80% of the epochs to obtain a smoother loss landscape to achieve faster convergence. The used losses were the same as the original papers, as described in section 4.1. We also used mixed precision training [182] to fit larger batches in the GPUs.

In addition, for the weight initialization, we used transfer learning in all experiments. However, in some branches we employed He Initialization (a.k.a Xavier Initialization) [183] on convolution-based modules of the decoder and Xavier Initialization (a.k.a Kaiming Initialization) [184] for the linear components. Table 10 lists all experiments carried out in this paper.

Table 10: Experiment summary. The displayed batch size is the batch size in each GPU. Since we used 3 GPUs, the global batch size is 3 × $batch\_size$. We chose different batch sizes in order to use all available GPU memory.

#	Method	Dataset	Batch Size	Epochs	Weight Initialization
1	ResNet-101 UNet FrameField [42]	AICrowd	80	100	Transfer learning from ImageNet pretrained weights on ResNet-101 backbone, Random He Initialization on UNet decoder, Segmentation Head and Frame Field Branch.
2	ResNet-101 UNet FrameField [42]	BAGS Buildings	80	100	Transfer learning from experiment #1 weights.
3	HRNet-w48-OCR FrameField	AICrowd	90	100	Transfer learning from CityScapes pretrained weights on HRNet-w48-OCR module, Random He Initialization on the segmentation head and Frame Field branch.
4	HRNet-w48-OCR FrameField	BAGS Buildings	90	100	Transfer learning from experiment #3 weights.
5	ResNet-101 DeepLabV3+ FrameField	AICrowd	100	100	Transfer learning from ImageNet pretrained weights on ResNet-101 backbone, Random He Initialization on the other parts of DeepLabV3+, segmentation head and Frame Field branch.
6	ResNet-101 DeepLabV3+ FrameField	BAGS Buildings	100	100	Transfer learning from #5.
7	ModPolyMapper [97]	AICrowd	20	20	Transfer learning from ImageNet pretrained weights for the object detection branch and Random He Initialization on PolygonRNN weights.
8	ModPolyMapper [97]	BAGS Buildings	20	20	From experiment #7 weights.

We trained our models using 3 NVIDIA Tesla V-100 GPUs, each with 32 Gb of RAM. The chosen framework for the experiments was PyTorch [129] version 1.10. We chose this framework due to the popularity shown in section 4, for the number of tutorials and documentation available online, and finally due to lots of papers on computer sciences that release their codes are also implemented in PyTorch and are a great teaching opportunity. One example of a paper that released their code and had a substantial influence on our research is [42], whose code is available at their GitHub repository[6].

Besides, we implemented a framework called pytorch_segmentation_models_trainer [43] and all

[6] https://github.com/Lydorn/Polygonization-by-Frame-Field-Learning

the experiment configurations and training weights are available in the GitHub repository[7] so that the research is fully reproducible.

4.4. Evaluation Metrics

To evaluate the vector outputs, we need first to match the corresponding features. [185] show several ways of performing feature matching, some based on distances, like Euclidean [186], Hausdorff [187] and Fréchet [188]. Another type of geometric matching described by [185] is the area overlap, also known as Intersection Over Union matching.

We matched the extracted polygons with the intersection over union metric, like in [97, 42], and computed the Polygons and Line Segments (PoLis) Metric [147] and the Mean Max Tangent Angle Errors (MMTAE) [42] in the matched polygons. Moreover, we evaluated the number of omissions (ratio of unmatched ground truth polygons and the total number of polygons) and excess (ratio of unmatched predicted polygons and the total number of polygons) of features that the neural networks extracted from the images. Finally, we computed the Intersection Over Union on the polygons of the whole dataset, shown in equation 1, considering the matched and unmatched features.

$$IoU = J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{\sum_{p_k \in A \cap B} area(p_k)}{\sum_{p_j \in A \cup B} area(p_j)} \tag{1}$$

where A and B are respectively, the ground truth and the predicted sets, $p_k \in A \cap B$ are the polygons in the intersection of A and B and $p_j \in A \cup B$ are the polygons in the union of A and B.

PoLiS ($p(A, B)$) is a vertex-by-vertex metric of similarity and it is defined by equation 2, where A and B are polygons such that $a_j \in A, j = 1, \ldots, q$ and $b_k \in B, k = 1, \ldots, r$ are vertexes from respectively A and B, $\partial A, \partial B$ are the boundaries of each polygon.

$$p(A, B) = \frac{1}{2q} \sum_{a_j \in A, b \in \partial B} \|a_j - b\| + \frac{1}{2r} \sum_{b_k \in B, a \in \partial A} \|b_k - a\| \tag{2}$$

Mean Max Tangent Angle Errors (MMTAE) is a metric to measure the regularity of the matched polygons. To compute this metric, first consider the predicted polygon P and the ground truth polygon Q. Sample n points in the boundary of P and for each point, find the closest point of the

[7] https://github.com/phborba/pytorch_segmentation_models_trainer

boundary Q. For each pair P_i and Q_i, compute the normalized tangent as shown in equations 3 and 4.

$$T(P_i) = \frac{P_{i+1} - P_i}{\|P_{i+1} - P_i\|} \tag{3}$$

$$T(Q_i) = \frac{Q_{i+1} - Q_i}{\|Q_{i+1} - Q_i\|} \tag{4}$$

After computing the normalized tangents, we can compute the metric (MMTAE) which defined by equation 5:

$$MMTAE = \max_{j \in V} \Delta\theta_j = \max_{j \in V} \left[\cos^{-1}\left(\langle T(P_i), T(Q_i) \rangle\right)\right] \tag{5}$$

where V is the set defined by $V = \{j \in [1, \ldots, n] \mid \frac{1}{2} < \frac{\|Q_{j+1} - Q_j\|}{\|P_{j+1} - P_j\|} < 2\}$, $\Delta\theta_j$ is the angle computed by the scalar product between the normalized tangents $T(P_j)$ (equation 3) and $T(Q_j)$ (equation 3).

5. Results

After conducting all experiments listed in table 10, we computed the test datasets' evaluation metrics described in subsection 4.4. On the one hand, the HRNet OCR W48 Frame Field method (experiment #3) presented better results on AICrowd dataset on all metrics but MTAE. On the other hand, the BAGS Buildings dataset's best method was ModPolyMapper (experiment #8), which had better results on all metrics. Table 11 shows the results of all carried out experiments.

Table 11: Results of each experiment.

Dataset	Method	PoLiS ↓	MMTAE ↓	IoU ↑	Omissions ↓	Excess ↓
AICrowd	HRNet OCR W48 Frame Field	**1.7**	43.89	**0.86**	**0.12**	**0.12**
	ResNet-101 UNet Frame Field	2.0	43.79	0.68	0.19	0.19
	ResNet-101 DeepLabV3+ Frame Field	2.13	45.5	0.78	0.16	0.16
	ModPolyMapper	2.48	**21.0**	0.69	0.16	0.16
BAGS	ModPolyMapper	**1.75**	**7.55**	**0.78**	**0.01**	**0.01**
	ResNet-101 UNet Frame Field	1.94	39.85	0.73	0.06	0.06
	ResNet-101 DeepLabV3+ Frame Field	2.22	39.91	0.7	0.1	0.1
	HRNet OCR W48 Frame Field	2.25	38.72	0.62	0.11	0.11

The visual results of each method on AICrowd test dataset can be seen in figures 17 and 18. On these figures, we can see some cases where the predicted polygons of HRNet OCR W48 Frame Field and ModPolyMapper are very similar to the Ground Truth. ModPolyMapper struggles with large buildings and complex building formats, but HRNet OCR W48 does not produce sharp right angles like ModPolyMapper.

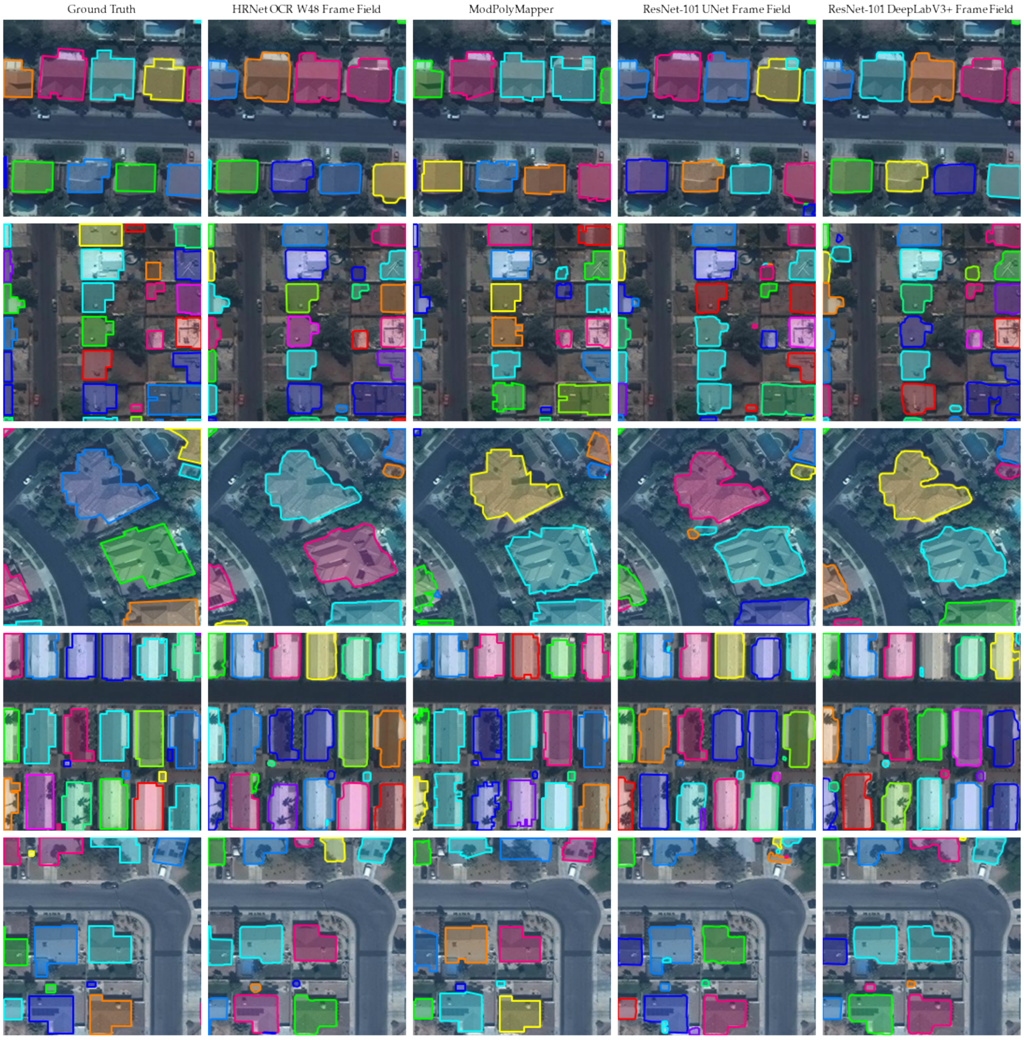

Ground Truth
HRNet OCR W48 Frame Field
ModPolyMapper
ResNet-101 UNet Frame Field
ResNet-101 DeepLabV3+ Frame Field

Figure 17: Visual results of the experiments on the AICrowd dataset.

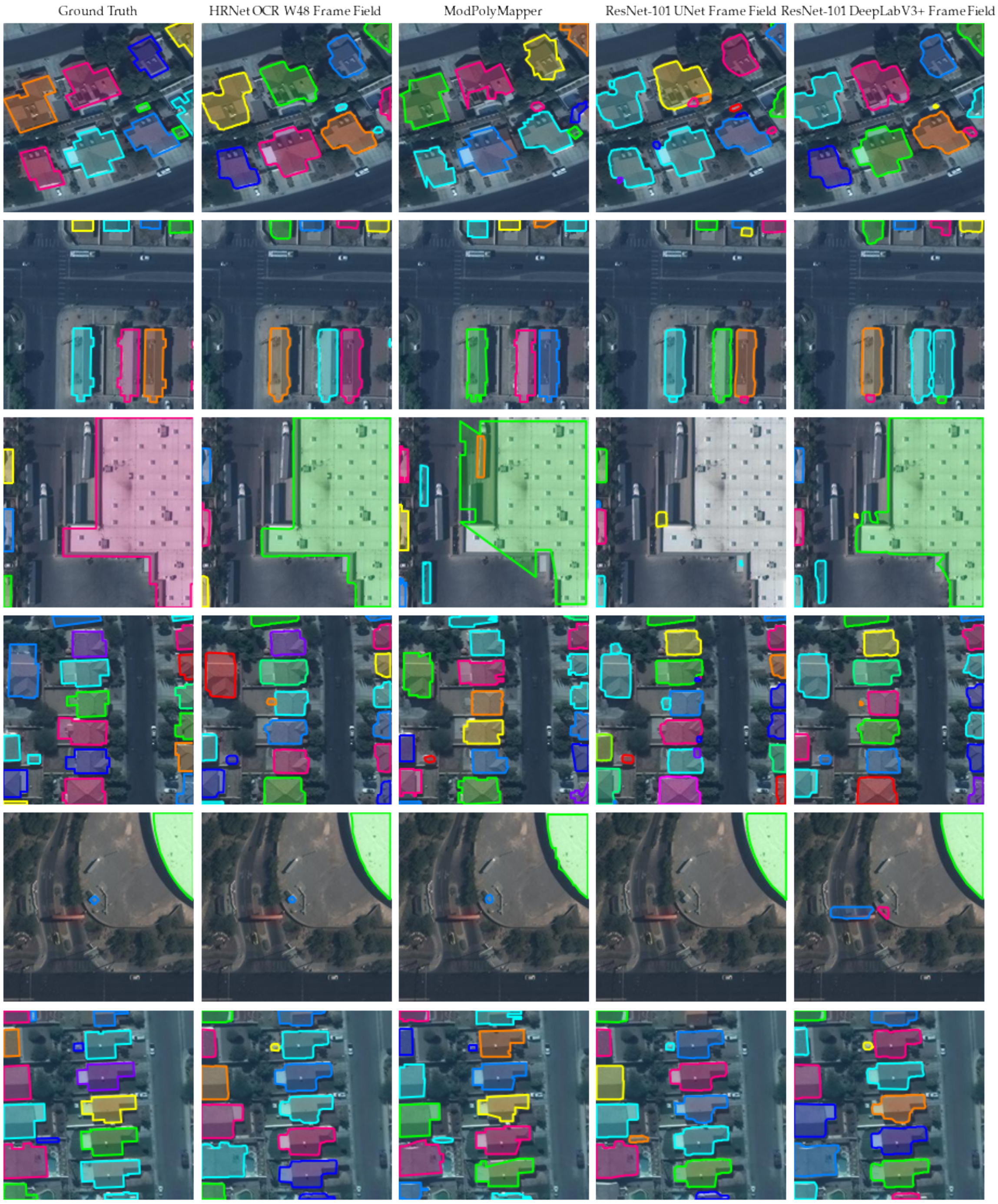

Ground Truth
HRNet OCR W48 Frame Field
ModPolyMapper
ResNet-101 UNet Frame Field
ResNet-101 DeepLabV3+ Frame Field

Figure 18: Visual results of the experiments on the AICrowd dataset.

In addition, the visual results of each method on BAGS Building dataset can be seen in figures 19 and 20. The number of omissions on ModPolyMapper outputs is visually larger on tiles with a high density of buildings, like seen in figure 20. 435

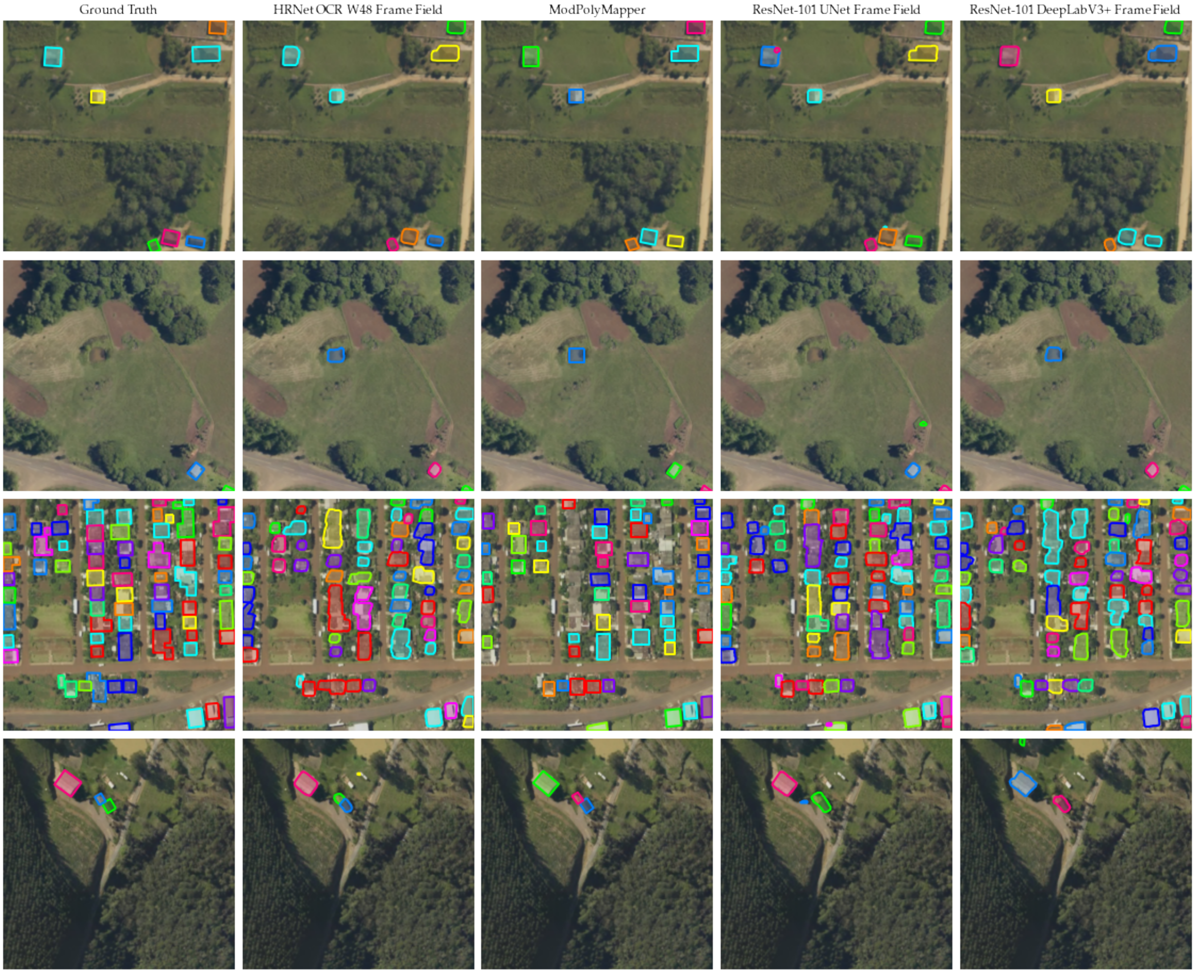

Ground Truth
HRNet OCR W48 Frame Field
ModPolyMapper
ResNet-101 UNet Frame Field
ResNet-101 DeepLabV3+ Frame Field

Figure 19: Visual results of the experiments in the BAGS Buildings dataset.

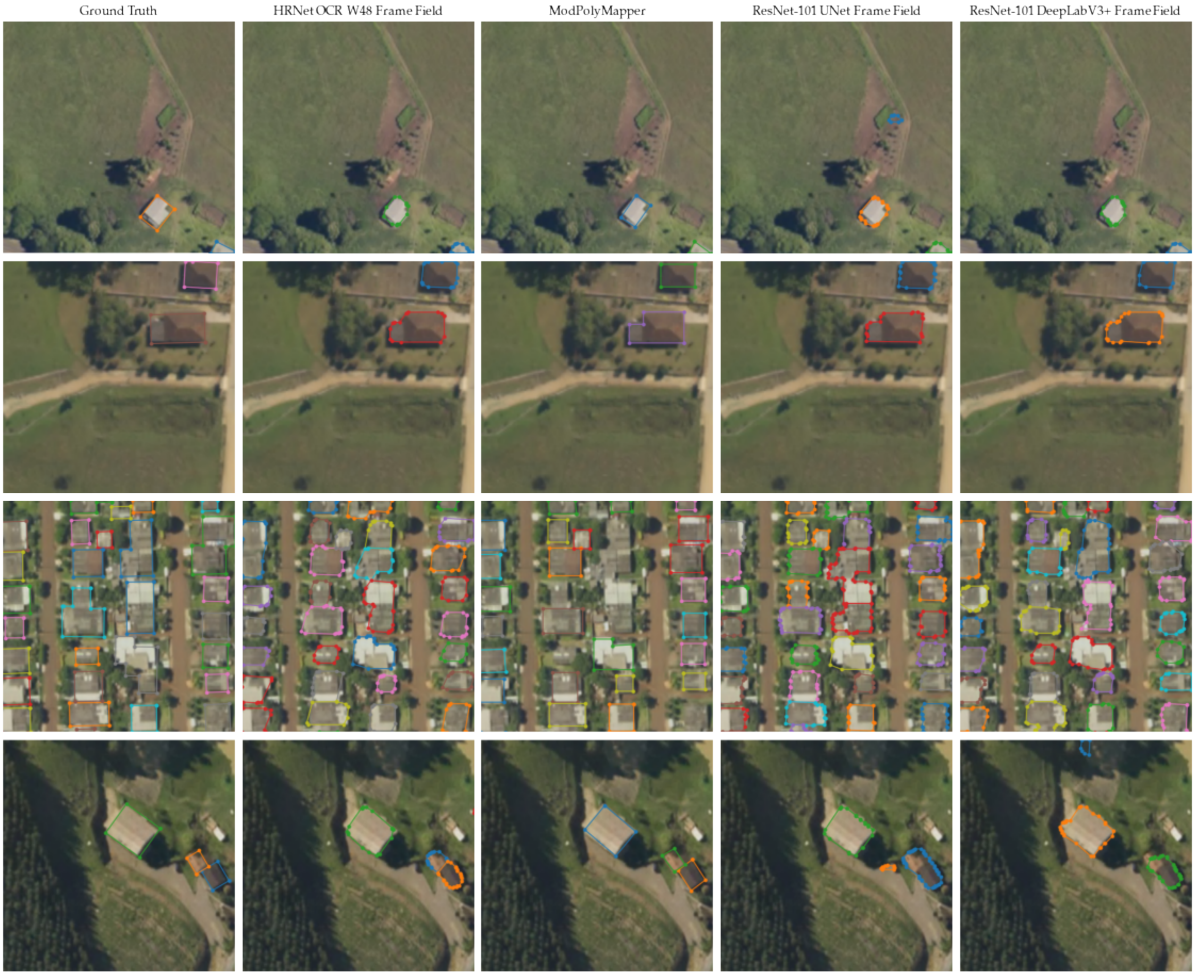

Ground Truth
HRNet OCR W48 Frame Field
ModPolyMapper
ResNet-101 UNet Frame Field
ResNet-101 DeepLabV3+ Frame Field

Figure 20: Visual results of the experiments in the BAGS Buildings dataset.

Furthermore, we investigated the behavior of HRNet OCR W48 Frame Field and ModPolyMapper on the BAGS Building dataset. We want to find out why the first method worked better on the AICrowd dataset, and its behavior was not the same as the BAGS Building dataset. After visually inspecting several output tiles of the BAGS Buildings dataset, we hypothesize that the first method works better on tiles with a high density of buildings and the second works better on sparse tiles.

Note that the AICrowd test dataset has a high building density, differently from the BAGS Building dataset, which has a majority of sparse tiles, like shown in the histogram in figure 21.

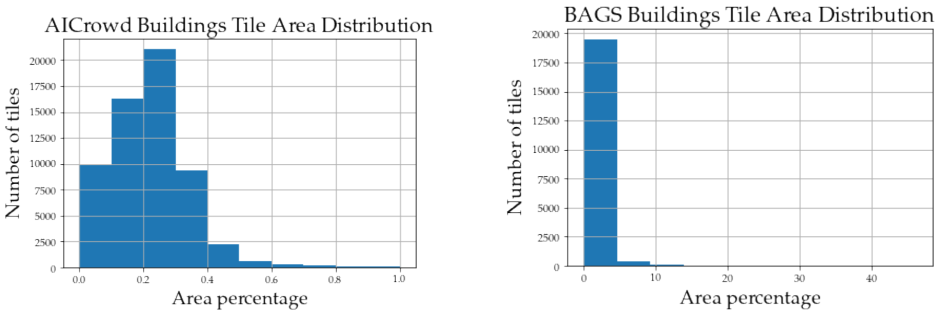


Figure 21: Histogram of the distribution of the total area of the buildings in a particular tile

Moreover, we related the average PoLiS of each tile on the test dataset with the ratio between the total area of the buildings and the total tile area. We also related the area ratio with the excess and the omission rate per tile. We want to identify whether PoLiS, excess ratio and omission ratio correlate with the area ratio. If there is a positive correlation, the higher the tile building density is, the worse the PoLiS value we would get. Similarly, a positive correlation between the omission rate or the excess rate with the area ratio might indicate a relation of the density of the tiles with these types of errors.

Before we test the correlation, we first need to assess if the area ratio and the average PoLiS have a normal distribution. We could not perform a Shapiro-Wilk test because the considered data has more than 5000 samples. So, to assess whether they follow a normal distribution, we perform a One-sample Kolmogorov-Smirnov test [189] on each variable, with a confidence interval of 95%, with the following hypothesis:

- H_0: The variable follow a normal distribution;
- H_1: The variable does not follow a normal distribution.

The test output and p-value for each variable are represented in table 12.

Table 12: One-sample Kolmogorov-Smirnov test result.

Variable	ModPolyMapper		HRNet OCR W48 Frame Field	
	Statistic (D)	p-value	Statistic (D)	p-value
Area ratio	1.0	2.2×10^{-16}	1.0	2.2×10^{-16}
PoLiS	0.71172	2.2×10^{-16}	0.81501	2.2×10^{-16}

Since the p-value on each test is small, we reject H_0 for each test and conclude that both area ratio and PoLiS variables do not follow a normal distribution. We need to test the normality of the excess and omission ratios. These two variables have ties (repeated values), so the one-sample Kolmogorov-Smirnov test is not well suited for this problem. Consequently, we proceeded with the Anderson-Darling test [190] for normality, with a confidence interval of 95% and the same hypothesis as before. Table 13 shows the results of the normality test.

Table 13: Anderson-Darling test result.

Variable	ModPolyMapper		HRNet OCR W48 Frame Field	
	Statistic (D)	p-value	Statistic (D)	p-value
Omission ratio	7652.3	2.2×10^{-16}	4451.3	2.2×10^{-16}
Excess ratio	7652.3	2.2×10^{-16}	4451.3	2.2×10^{-16}

Therefore, since the p-value on each test is small, we reject H_0 for each test and conclude that both omission ratio and excess ratio variables do not follow a normal distribution. Thus, to test the correlation between all the chosen variables, we need to use a non-parametric method such as Kendall's rank correlation tau [191], with a confidence interval of 95%, with the following hypothesis for each pair of comparisons:

- H_0: There is no association between the variables;
- H_1: There is an association between the variables.

Table 14 shows the results of the all tests on both methods. The p-value on all tests is small, so we reject H_0 and have an association for each pair of variables.

Table 14: Kendall's rank correlation tau result.

Data	ModPolyMapper		HRNet OCR W48 Frame Field	
	tau (τ)	p-value	tau (τ)	p-value
Area ratio and Polis	0.2702111	2.2×10^{-16}	0.2645045	2.2×10^{-16}
Area ratio and omissions ratio	0.02951534	3.244×10^{-7}	0.09434868	2.2×10^{-16}
Area ratio and excess ratio	0.02951534	3.244×10^{-7}	0.09434868	2.2×10^{-16}

Since the p-value is small, according to [190, 192, 193], there is a moderate correlation between the area ratio and the PoLiS on both methods, but the correlation is stronger on ModPolyMapper than HRNet OCR w48 Frame Field. Similarly, there is a weak correlation between the area and the variables excess and omission ratios. Thus, we conclude that there is evidence that on the BAGS Building dataset, ModPolyMapper works better on sparse tiles, and the HRNet OCR W48 Frame Field method works better on dense tiles.

6. Conclusion

This paper carried out a meta-analysis on remote sensing papers about Building Footprint Extraction using deep learning techniques, focusing on techniques that can extract the polygons in vector format. We identified related research problems, the techniques employed, and the top 10 papers in citations. We also identified the PyTorch as the most popular framework among the selected research, followed closely by Tensorflow. We also concluded that very few papers (13 out of 99) provide their codes, which may show that some researchers of the Remote Sensing field could adopt open science habits like sharing the code, sharing the data used and sharing the trained neural network weights, so that their research is fully reproducible.

Furthermore, we narrowed our research and analyzed only Building Footprint Extraction papers, identifying that the most popular datasets are the WHU Aerial Building dataset, followed by the Massachusetts Building dataset and the INRIA Aerial Labeling dataset. In addition, SpaceNet Building, CrowdAI, and Open Cities are examples of datasets that release the imagery and the raster building masks and release the vector data that originated the masks. Moreover, we concluded that the most popular deep learning architecture employed in the studied papers is the U-Net, and the most popular backbones are from the ResNet family. Thus, we identified that most papers use only RGB imagery as input and outputs only raster masks.

Then, we selected the papers whose output is in vector format and analyzed them in terms of inputs types, datasets used, architecture and backbone chosen, loss function, data augmentation, and evaluation metrics used. Among these papers, we selected two: [78, 97], which represent end-to-end methods built on top of deep convolutional neural networks.

On top of that, we performed a survey on computer sciences methods to identify other suitable candidates. In this search, we came across methods like PolygonRNN [152], PolygonRNN++ [154], DARNet [153], Curve-GCN [155], Conv-MPN [156], PolyMapper [161], and Polygonization by Frame Field [42].

After the review, we chose Polygonization by Frame Field [42] and the ModPolyMapper [97] to carry out our experiments. We proposed the Brazilian Army Geographic Service (BAGS) Buildings dataset, with rural and urban scenes from the Brazilian states of Rio Grande do Sul and Santa Catarina. We also proposed variations of Polygonization by Frame Field, one based on HRNet OCR W48 and the other based on DeepLabV3+. We presented our framework, pytorch_segmentation_model_trainer, and released our research code online, the configuration files used in the training procedures, and the neural network weights obtained in the experiments of this research.

We carried out the experiments with the parameters defined in section 4 and concluded that our method HRNet OCR W48 Frame Field outperforms all others in the AICrowd dataset, but it does not work well in the BAGS Buildings dataset. For this last dataset, the best result was ModPolyMapper, different from the results on the AICrowd dataset. We hypothesized that this difference in the results might be connected with the different nature of both datasets: AICrowd represents an urban environment, while the BAGS Buildings dataset is in its majority sparse. We investigated this assumption with statistical hypothesis tests and correlation tests and concluded that the difference in the datasets could explain this discrepancy in the results. We concluded in our experiments that ModPolyMapper works better on sparse areas, while HRNet OCR W48 Frame Field can produce better results in dense areas. We suggest further research on this matter by mixing up different datasets and performing new training.

Besides, we also noted that parameter tunning might lead to better results since there are many parameters in the ASM proposed by [42]. So we suggest further research in a new combination of parameters so that polygons with more regular formats and more similar to the ground truth polygons can be extracted. An exciting research opportunity is investigating the overlap rate of the extracted polygons of ModPolyMapper and ways to fix them so that this method can be used for cartographic purposes.

Additionally, new research on the usage of attention mechanisms like Convolutional Block Attention Module (CBAM) [167] or Polarized Self Attention [194] coupled to HRNet OCR W48 Frame Field may be interesting to assess whether these structures could improve the results of this method.

In addition, research on the inference methods may also be interesting. The results could be improved if we used a sliding window in either mask-based methods such as Frame Field or object detection methods like ModPolyMapper. Particularly in the experiments involving object detection, we propose research using the Slicing Aided Hyper Inference technique implemented in the python package Sahi [195]. This technique improves the detection of small objects in large images and could improve the results of ModPolyMapper.

Withal, [164] proposed a novel method named Deep Snake, which takes a contour produced from a CNN backbone as input and produces a vertex by vertex inference around the selected object. It consists of three blocks, a CNN backbone, a fusion block, and a prediction block. It presented thrilling results and released its code online, enabling quicker research. Thou, we also suggest researching a neighborhood mechanism to deal with adjacent features so that this new method could mitigate gaps and overlaps that may occur.

Finally, with the recent advances on vision transformers, we suggest the research of backbones like SWIN Transformer V2 [196] as new backbones of both ModPolyMapper and Frame Field methods.

References

[1] H. Arefi, P. Reinartz, Building reconstruction using dsm and orthorectified images, Remote Sensing 5 (4) (2013) 1681–1703. doi:10.3390/rs5041681.
URL https://www.mdpi.com/2072-4292/5/4/1681

[2] M. Awrangjeb, S. A. N. Gilani, F. U. Siddiqui, An effective data-driven method for 3-d building roof reconstruction and robust change detection, Remote Sensing 10 (10). doi:10.3390/rs10101512.
URL https://www.mdpi.com/2072-4292/10/10/1512

[3] D. Poli, I. Caravaggi, 3d modeling of large urban areas with stereo vhr satellite imagery: lessons learned, Natural hazards 68 (1) (2013) 53–78.

[4] E. Tarantino, B. Figorito, Extracting buildings from true color stereo aerial images using a decision making strategy, Remote Sensing 3 (8) (2011) 1553–1567. doi:10.3390/rs3081553.
URL https://www.mdpi.com/2072-4292/3/8/1553

[5] Y. Hsieh, D. McKeown, F. Perlant, Performance evaluation of scene registration and stereo matching for cartographic feature extraction, IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (2) (1992) 214–238. doi:10.1109/34.121790.

[6] C. Baillard, H. Maıtre, 3-d reconstruction of urban scenes from aerial stereo imagery: A focusing strategy, Computer Vision and Image Understanding 76 (3) (1999) 244–258. doi:https://doi.org/10.1006/cviu.1999.0793.
URL https://www.sciencedirect.com/science/article/pii/S1077314299907932

[7] B. Sirmacek, H. Taubenbock, P. Reinartz, M. Ehlers, Performance evaluation for 3-d city model generation of six different dsms from air- and spaceborne sensors, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 5 (1) (2012) 59–70. doi:10.1109/JSTARS.2011.2178399.

[8] S. Ji, S. Wei, M. Lu, Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set, IEEE Transactions on Geoscience and Remote Sensing 57 (1) (2019) 574–586. doi:10.1109/TGRS.2018.2858817.

[9] S. Ji, S. Wei, M. Lu, A scale robust convolutional neural network for automatic building extraction from aerial and satellite imagery, International Journal of Remote Sensing 40 (9) (2019) 3308–3322. arXiv:https://doi.org/10.1080/01431161.2018.1528024, doi:10.1080/01431161.2018.1528024.
URL https://doi.org/10.1080/01431161.2018.1528024

[10] Y. Yi, Z. Zhang, W. Zhang, C. Zhang, W. Li, T. Zhao, Semantic segmentation of urban buildings from vhr remote sensing imagery using a deep convolutional neural network, Remote Sensing 11 (15). doi:10.3390/rs11151774.
URL https://www.mdpi.com/2072-4292/11/15/1774

[11] P. Liu, X. Liu, M. Liu, Q. Shi, J. Yang, X. Xu, Y. Zhang, Building footprint extraction from high-resolution images via spatial residual inception convolutional neural network, Remote Sensing 11 (7). doi:10.3390/rs11070830.
URL https://www.mdpi.com/2072-4292/11/7/830

[12] H. Yang, P. Wu, X. Yao, Y. Wu, B. Wang, Y. Xu, Building extraction in very high resolution imagery by dense-attention networks, Remote Sensing 10 (11). doi:10.3390/rs10111768.
URL https://www.mdpi.com/2072-4292/10/11/1768

[13] Z. Shao, P. Tang, Z. Wang, N. Saleem, S. Yam, C. Sommai, Brrnet: A fully convolutional neural network for automatic building extraction from high-resolution remote sensing images, Remote Sensing 12 (6). doi:10.3390/rs12061050.
URL https://www.mdpi.com/2072-4292/12/6/1050

[14] T. Lu, D. Ming, X. Lin, Z. Hong, X. Bai, J. Fang, Detecting building edges from high spatial resolution remote sensing imagery using richer convolution features network, Remote Sensing 10 (9). doi:10.3390/rs10091496.
URL https://www.mdpi.com/2072-4292/10/9/1496

[15] L. Li, J. Liang, M. Weng, H. Zhu, A multiple-feature reuse network to extract buildings from remote sensing imagery, Remote Sensing 10 (9). doi:10.3390/rs10091350.
URL https://www.mdpi.com/2072-4292/10/9/1350

[16] S. Ji, Y. Shen, M. Lu, Y. Zhang, Building instance change detection from large-scale aerial

images using convolutional neural networks and simulated samples, Remote Sensing 11 (11) (2019) 1343.

[17] Y. Liu, C. Pang, Z. Zhan, X. Zhang, X. Yang, Building change detection for remote sensing images using a dual-task constrained deep siamese convolutional network model, IEEE Geoscience and Remote Sensing Letters 18 (5) (2020) 811–815.

[18] D. Peng, L. Bruzzone, Y. Zhang, H. Guan, H. Ding, X. Huang, Semicdnet: A semisupervised convolutional neural network for change detection in high resolution remote-sensing images, IEEE Transactions on Geoscience and Remote Sensing.

[19] Y. Bai, C. Gao, S. Singh, M. Koch, B. Adriano, E. Mas, S. Koshimura, A framework of rapid regional tsunami damage recognition from post-event terrasar-x imagery using deep neural networks, IEEE Geoscience and Remote Sensing Letters 15 (1) (2018) 43–47. `doi:10.1109/LGRS.2017.2772349`.

[20] D. Duarte, F. Nex, N. Kerle, G. Vosselman, Multi-resolution feature fusion for image classification of building damages with convolutional neural networks, Remote Sensing 10 (10). `doi:10.3390/rs10101636`.
URL `https://www.mdpi.com/2072-4292/10/10/1636`

[21] S. Ghaffarian, N. Kerle, E. Pasolli, J. Jokar Arsanjani, Post-disaster building database updating using automated deep learning: An integration of pre-disaster openstreetmap and multi-temporal satellite data, Remote Sensing 11 (20). `doi:10.3390/rs11202427`.
URL `https://www.mdpi.com/2072-4292/11/20/2427`

[22] F. Nex, D. Duarte, F. G. Tonolo, N. Kerle, Structural building damage detection with deep learning: Assessment of a state-of-the-art cnn in operational conditions, Remote Sensing 11 (23). `doi:10.3390/rs11232765`.
URL `https://www.mdpi.com/2072-4292/11/23/2765`

[23] W. Yang, X. Zhang, P. Luo, Transferability of convolutional neural network models for identifying damaged buildings due to earthquake, Remote Sensing 13 (3). `doi:10.3390/rs13030504`.
URL `https://www.mdpi.com/2072-4292/13/3/504`

[24] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, B. A. Johnson, Deep learning in remote sensing applications: A meta-analysis and review, ISPRS Journal of Photogrammetry and Remote Sensing 152 (March) (2019) 166–177. doi:10.1016/j.isprsjprs.2019.04.015.
URL https://doi.org/10.1016/j.isprsjprs.2019.04.015

[25] T. Hoeser, C. Kuenzer, Object detection and image segmentation with deep learning on Earth observation data: A review-part I: Evolution and recent trends, Remote Sensing 12 (10). doi:10.3390/rs12101667.

[26] T. Hoeser, F. Bachofer, C. Kuenzer, Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review—Part II: Applications, Remote Sensing 12 (18) (2020) 3053. doi:10.3390/rs12183053.

[27] B. Neupane, T. Horanont, J. Aryal, Deep learning-based semantic segmentation of urban features in satellite images: A review and meta-analysis, Remote Sensing 13 (2021) 1–41. doi:10.3390/rs13040808.

[28] D. Holland, D. Boyd, P. Marshall, Updating topographic mapping in great britain using imagery from high-resolution satellite sensors, ISPRS Journal of Photogrammetry and Remote Sensing 60 (3) (2006) 212–223, extraction of Topographic Information from High-Resolution Satellite Imagery. doi:https://doi.org/10.1016/j.isprsjprs.2006.02.002.
URL https://www.sciencedirect.com/science/article/pii/S0924271606000086

[29] A. G. Guimarães Filho, P. Borba, Methodology for land mapping of amapa state-a special case of amazon radiography project, in: IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium, 2020, pp. 1540–1543. doi:10.1109/IGARSS39084.2020.9324673.

[30] A. G. G. Filho, P. Borba, V. H. S. Silva, A. Cerdeira, A. P. D. Poz, Quality control relevance on acquisition of large scale geospatial data to urban territorial management, in: 2020 IEEE Latin American GRSS ISPRS Remote Sensing Conference (LAGIRS), 2020, pp. 138–142. doi:10.1109/LAGIRS48042.2020.9165682.

[31] C. Duchêne, B. Baella, C. Brewer, D. Burghardt, B. Buttenfield, J. Gaffuri, D. Käuferle, F. Lecordix, E. Maugeais, R. Nijhuis, M. Pla, M. Post, N. Regnauld, L. Stanislawski, J. Stoter,

K. Tóth, S. Urbanke, V. van Altena, A. Wiedemann, Generalisation in Practice Within National Mapping Agencies, Springer, 2014, pp. 329–391. doi:10.1007/978-3-319-00203-3_11.

[32] M. F. Goodchild, Citizens as sensors: the world of volunteered geography, GeoJournal 69 (4) (2007) 211–221.

[33] J. E. Vargas-Munoz, S. Srivastava, D. Tuia, A. X. Falcão, Openstreetmap: Challenges and opportunities in machine learning and remote sensing, IEEE Geoscience and Remote Sensing Magazine 9 (1) (2021) 184–199. doi:10.1109/MGRS.2020.2994107.

[34] J. Chen, Y. Zhou, A. Zipf, H. Fan, Deep learning from multiple crowds: A case study of humanitarian mapping, IEEE Transactions on Geoscience and Remote Sensing 57 (3) (2019) 1713–1722. doi:10.1109/TGRS.2018.2868748.

[35] H. Li, M. Hu, Y. Huang, Automatic identification of overpass structures: A method of deep learning, ISPRS International Journal of Geo-Information 8 (9). doi:10.3390/ijgi8090421.
URL https://www.mdpi.com/2220-9964/8/9/421

[36] J. E. Vargas Munoz, D. Tuia, A. X. Falcão, Deploying machine learning to assist digital humanitarians: making image annotation in openstreetmap more efficient, International Journal of Geographical Information Science 35 (9) (2021) 1725–1745.

[37] Y. Xu, Z. Chen, Z. Xie, L. Wu, Quality assessment of building footprint data using a deep autoencoder network, International Journal of Geographical Information Science 31 (10) (2017) 1929–1951.

[38] Q. T. Truong, G. Touya, C. d. Runz, Osmwatchman: Learning how to detect vandalized contributions in osm using a random forest classifier, ISPRS International Journal of Geo-Information 9 (9). doi:10.3390/ijgi9090504.
URL https://www.mdpi.com/2220-9964/9/9/504

[39] D. Bonafilia, J. Gill, S. Basu, D. Yang, Building high resolution maps for humanitarian aid and development with weakly- and semi-supervised learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2019.

[40] W. Sirko, S. Kashubin, M. Ritter, A. Annkah, Y. S. E. Bouchareb, Y. Dauphin, D. Keysers, M. Neumann, M. Cisse, J. Quinn, Continental-scale building detection from high resolution satellite imagery (2021). `arXiv:2107.12283`.

[41] E. L. Usery, S. T. Arundel, E. Shavers, L. Stanislawski, P. Thiem, D. Varanka, Geoai in the us geological survey for topographic mapping, Transactions in GIS.

[42] N. Girard, D. Smirnov, J. Solomon, Y. Tarabalka, Polygonal building extraction by frame field learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 5891–5900.

[43] P. Borba, phborba/segmentation_models_trainer: First Release (sep 2020). `doi:10.5281/zenodo.4060390`.
URL `https://doi.org/10.5281/zenodo.4060390`

[44] A. Abdollahi, B. Pradhan, A. M. Alamri, An ensemble architecture of deep convolutional segnet and unet networks for building semantic segmentation from high-resolution aerial images, Geocarto International (2020) 1–16.

[45] J. Chen, F. He, Y. Zhang, G. Sun, M. Deng, Spmf-net: Weakly supervised building segmentation by combining superpixel pooling and multi-scale feature fusion, Remote Sensing 12 (6) (2020) 1049.

[46] S. He, W. Jiang, Boundary-assisted learning for building extraction from optical remote sensing imagery, Remote Sensing 13 (4) (2021) 760.

[47] N. Yang, H. Tang, Semantic segmentation of satellite images: A deep learning approach integrated with geospatial hash codes, Remote Sensing 13 (14). `doi:10.3390/rs13142723`.
URL `https://www.mdpi.com/2072-4292/13/14/2723`

[48] H. Liu, J. Luo, B. Huang, X. Hu, Y. Sun, Y. Yang, N. Xu, N. Zhou, De-net: Deep encoding network for building extraction from high-resolution remote sensing imagery, Remote Sensing 11 (20). `doi:10.3390/rs11202380`.
URL `https://www.mdpi.com/2072-4292/11/20/2380`

[49] Y. Wang, G. Wu, Y. Guo, Y. Huang, R. Shibasaki, Learn to extract building outline from misaligned annotation through nearest feature selector, Remote Sensing 12 (17) (2020) 2722.

[50] Q. Zhu, C. Liao, H. Hu, X. Mei, H. Li, Map-net: Multiple attending path neural network for building footprint extraction from remote sensed imagery, IEEE Transactions on Geoscience and Remote Sensing 59 (7) (2021) 6169–6181. doi:10.1109/TGRS.2020.3026051.

[51] X. Wei, X. Li, W. Liu, L. Zhang, D. Cheng, H. Ji, W. Zhang, K. Yuan, Building outline extraction directly using the u2-net semantic segmentation model from high-resolution aerial images and a comparison study, Remote Sensing 13 (16) (2021) 3187.

[52] X. Li, X. Yao, Y. Fang, Building-a-nets: Robust building extraction from high-resolution remote sensing images with adversarial networks, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (10) (2018) 3680–3687. doi:10.1109/JSTARS.2018.2865187.

[53] H. Guo, Q. Shi, B. Du, L. Zhang, D. Wang, H. Ding, Scene-driven multitask parallel attention network for building extraction in high-resolution remote sensing images, IEEE Transactions on Geoscience and Remote Sensing 59 (5) (2021) 4287–4306. doi:10.1109/TGRS.2020.3014312.

[54] A. Milosavljević, Automated processing of remote sensing imagery using deep semantic segmentation: A building footprint extraction case, ISPRS International Journal of Geo-Information 9 (8) (2020) 486.

[55] T. Zhang, H. Tang, Y. Ding, P. Li, C. Ji, P. Xu, Fsrss-net: High-resolution mapping of buildings from middle-resolution satellite images using a super-resolution semantic segmentation network, Remote Sensing 13 (12) (2021) 2290.

[56] Y. Zhang, W. Gong, J. Sun, W. Li, Web-net: A novel nest networks with ultra-hierarchical sampling for building extraction from aerial imageries, Remote Sensing 11 (16). doi:10.3390/rs11161897.
URL https://www.mdpi.com/2072-4292/11/16/1897

[57] S. Guo, Q. Jin, H. Wang, X. Wang, Y. Wang, S. Xiang, Learnable gated convolutional neural network for semantic segmentation in remote-sensing images, Remote Sensing 11 (16) (2019) 1922.

[58] Y. Zhang, W. Li, W. Gong, Z. Wang, J. Sun, An improved boundary-aware perceptual loss for building extraction from vhr images, Remote Sensing 12 (7) (2020) 1195.

[59] J. Liu, S. Wang, X. Hou, W. Song, A deep residual learning serial segmentation network for extracting buildings from remote sensing imagery, International Journal of Remote Sensing 41 (14) (2020) 5573–5587.

[60] J. Huang, X. Zhang, Y. Sun, Q. Xin, Attention-guided label refinement network for semantic segmentation of very high resolution aerial orthoimages, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 4490–4503. `doi:10.1109/JSTARS.2021.3073935`.

[61] Y. Li, W. Xu, H. Chen, J. Jiang, X. Li, A novel framework based on mask r-cnn and histogram thresholding for scalable segmentation of new and old rural buildings, Remote Sensing 13 (6) (2021) 1070.

[62] Q. Yuan, H. Z. Mohd Shafri, A. H. Alias, S. J. b. Hashim, Multiscale semantic feature optimization and fusion network for building extraction using high-resolution aerial images and lidar data, Remote Sensing 13 (13) (2021) 2473.

[63] W. Feng, H. Sui, L. Hua, C. Xu, G. Ma, W. Huang, Building extraction from vhr remote sensing imagery by combining an improved deep convolutional encoder-decoder architecture and historical land use vector map, International Journal of Remote Sensing 41 (17) (2020) 6595–6617.

[64] J. Huang, X. Zhang, Q. Xin, Y. Sun, P. Zhang, Automatic building extraction from high-resolution aerial images and lidar data using gated residual refinement network, ISPRS Journal of Photogrammetry and Remote Sensing 151 (2019) 91–105. `doi:https://doi.org/10.1016/j.isprsjprs.2019.02.019`.
URL `https://www.sciencedirect.com/science/article/pii/S0924271619300590`

[65] H. T. Aung, S. H. Pha, W. Takeuchi, Building footprint extraction in yangon city from monocular optical satellite image using deep learning, Geocarto International 0 (0) (2020) 1–21. `arXiv:https://doi.org/10.1080/10106049.2020.1740949`, `doi:10.1080/10106049.2020.1740949`.
URL `https://doi.org/10.1080/10106049.2020.1740949`

[66] L. Xia, X. Zhang, J. Zhang, H. Yang, T. Chen, Building extraction from very-high-resolution remote sensing images using semi-supervised semantic edge detection, Remote Sensing 13 (11) (2021) 2187.

[67] P. Xu, H. Tang, J. Ge, L. Feng, Espc_nasunet: An end-to-end super-resolution semantic segmentation network for mapping buildings from remote sensing images, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[68] R. Davari Majd, M. Momeni, P. Moallem, Transferable object-based framework based on deep convolutional neural networks for building extraction, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12 (8) (2019) 2627–2635. doi:10.1109/JSTARS.2019.2924582.

[69] W. Li, C. He, J. Fang, J. Zheng, H. Fu, L. Yu, Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source GIS data, Remote Sensing 11 (4). doi:10.3390/rs11040403.

[70] E. Maltezos, A. Doulamis, N. Doulamis, C. Ioannidis, Building extraction from lidar data applying deep convolutional neural networks, IEEE Geoscience and Remote Sensing Letters 16 (1) (2019) 155–159. doi:10.1109/LGRS.2018.2867736.

[71] K. Rastogi, P. Bodani, S. A. Sharma, Automatic building footprint extraction from very high-resolution imagery using deep learning techniques, Geocarto International (2020) 1–13.

[72] P. Zhang, P. Du, C. Lin, X. Wang, E. Li, Z. Xue, X. Bai, A hybrid attention-aware fusion network (hafnet) for building extraction from high-resolution imagery and lidar data, Remote Sensing 12 (22) (2020) 3764.

[73] Y. Liu, D. Chen, A. Ma, Y. Zhong, F. Fang, K. Xu, Multiscale u-shaped cnn building instance extraction framework with edge constraint for high-spatial-resolution remote sensing imagery, IEEE Transactions on Geoscience and Remote Sensing.

[74] D.-Y. Chen, L. Peng, W.-C. Li, Y.-D. Wang, Building extraction and number statistics in wui areas based on unet structure and ensemble learning, Remote Sensing 13 (6) (2021) 1172.

[75] J. Cai, Y. Chen, Mha-net: Multipath hybrid attention network for building footprint extraction from high-resolution remote sensing imagery, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[76] K. Bittner, F. Adam, S. Cui, M. Körner, P. Reinartz, Building footprint extraction from vhr remote sensing images combined with normalized dsms using fused fully convolutional networks, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (8) (2018) 2615–2629. `doi:10.1109/JSTARS.2018.2849363`.

[77] P. Schuegraf, K. Bittner, Automatic building footprint extraction from multi-resolution remote sensing images using a hybrid fcn, ISPRS International Journal of Geo-Information 8 (4). `doi:10.3390/ijgi8040191`.
URL `https://www.mdpi.com/2220-9964/8/4/191`

[78] Q. Chen, L. Wang, S. L. Waslander, X. Liu, An end-to-end shape modeling framework for vectorized building outline generation from aerial images, ISPRS Journal of Photogrammetry and Remote Sensing 170 (2020) 114–126.

[79] Z. Zhang, W. Guo, M. Li, W. Yu, Gis-supervised building extraction with label noise-adaptive fully convolutional neural network, IEEE Geoscience and Remote Sensing Letters 17 (12) (2020) 2135–2139.

[80] J. Hui, M. Du, X. Ye, Q. Qin, J. Sui, Effective building extraction from high-resolution remote sensing images with multitask driven deep neural network, IEEE Geoscience and Remote Sensing Letters 16 (5) (2018) 786–790.

[81] S. Ran, X. Gao, Y. Yang, S. Li, G. Zhang, P. Wang, Building multi-feature fusion refined network for building extraction from high-resolution remote sensing images, Remote Sensing 13 (14). `doi:10.3390/rs13142794`.
URL `https://www.mdpi.com/2072-4292/13/14/2794`

[82] L. Zhang, R. Dong, S. Yuan, W. Li, J. Zheng, H. Fu, Making low-resolution satellite images reborn: A deep learning approach for super-resolution building extraction, Remote Sensing 13 (15) (2021) 2872.

[83] C. Lin, S. Guo, J. Chen, L. Sun, X. Zheng, Y. Yang, Y. Xiong, Deep learning network intensification for preventing noisy-labeled samples for remote sensing classification, Remote Sensing 13 (9) (2021) 1689.

[84] Y. Xie, J. Zhu, Y. Cao, D. Feng, M. Hu, W. Li, Y. Zhang, L. Fu, Refined extraction of building outlines from high-resolution remote sensing imagery based on a multifeature convolutional neural network and morphological filtering, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13 (2020) 1842–1855.

[85] F. Shi, T. Zhang, A multi-task network with distance–mask–boundary consistency constraints for building extraction from aerial images, Remote Sensing 13 (14) (2021) 2656.

[86] W. Boonpook, Y. Tan, B. Xu, Deep learning-based multi-feature semantic segmentation in building extraction from images of uav photogrammetry, International Journal of Remote Sensing 42 (1) (2021) 1–19.

[87] Q. Li, Y. Shi, S. Auer, R. Roschlaub, K. Möst, M. Schmitt, C. Glock, X. Zhu, Detection of undocumented building constructions from official geodata using a convolutional neural network, Remote Sensing 12 (21) (2020) 3537.

[88] Y. Xu, L. Wu, Z. Xie, Z. Chen, Building extraction in very high resolution remote sensing imagery using deep learning and guided filters, Remote Sensing 10 (1). `doi:10.3390/rs10010144`.
URL `https://www.mdpi.com/2072-4292/10/1/144`

[89] G. Yang, Q. Zhang, G. Zhang, Eanet: Edge-aware network for the extraction of buildings from aerial images, Remote Sensing 12 (13) (2020) 2161.

[90] C. Liao, H. Hu, H. Li, X. Ge, M. Chen, C. Li, Q. Zhu, Joint learning of contour and structure for boundary-preserved building extraction, Remote Sensing 13 (6). `doi:10.3390/rs13061049`.
URL `https://www.mdpi.com/2072-4292/13/6/1049`

[91] Z. Chen, D. Li, W. Fan, H. Guan, C. Wang, J. Li, Self-attention in reconstruction bias u-net for semantic segmentation of building rooftops in optical remote sensing images, Remote Sensing 13 (13) (2021) 2524.

[92] S. Seong, J. Choi, Semantic segmentation of urban buildings using a high-resolution network (hrnet) with channel and spatial attention gates, Remote Sensing 13 (16) (2021) 3087.

[93] T. Wu, Y. Hu, L. Peng, R. Chen, Improved anchor-free instance segmentation for building extraction from high-resolution remote sensing images, Remote Sensing 12 (18) (2020) 2910.

[94] X. Pan, F. Yang, L. Gao, Z. Chen, B. Zhang, H. Fan, J. Ren, Building extraction from high-resolution aerial imagery using a generative adversarial network with spatial and channel attention mechanisms, Remote Sensing 11 (8). doi:10.3390/rs11080917.
URL https://www.mdpi.com/2072-4292/11/8/917

[95] W. Deng, Q. Shi, J. Li, Attention-gate-based encoder–decoder network for automatical building extraction, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 2611–2620.

[96] D. Wierzbicki, O. Matuk, E. Bielecka, Polish cadastre modernization with remotely extracted buildings from high-resolution aerial orthoimagery and airborne lidar, Remote Sensing 13 (4) (2021) 611.

[97] W. Zhao, C. Persello, A. Stein, Building outline delineation: From aerial images to polygons with an improved end-to-end learning framework, ISPRS Journal of Photogrammetry and Remote Sensing 175 (2021) 119–131. doi:https://doi.org/10.1016/j.isprsjprs.2021.02.014.
URL https://www.sciencedirect.com/science/article/pii/S0924271621000551

[98] C. Ding, L. Weng, M. Xia, H. Lin, Non-local feature search network for building and road segmentation of remote sensing image, ISPRS International Journal of Geo-Information 10 (4) (2021) 245.

[99] H. Guo, Q. Shi, A. Marinoni, B. Du, L. Zhang, Deep building footprint update network: A semi-supervised method for updating existing building footprint from bi-temporal remote sensing images, Remote Sensing of Environment 264 (2021) 112589. doi:https://doi.org/10.1016/j.rse.2021.112589.
URL https://www.sciencedirect.com/science/article/pii/S0034425721003096

[100] D. Griffiths, J. Boehm, Improving public data for building segmentation from convolutional neural networks (cnns) for fused airborne lidar and image data using active contours, ISPRS Journal of Photogrammetry and Remote Sensing 154 (2019) 70–83. doi:https://doi.org/10.1016/j.isprsjprs.2019.05.013.
URL https://www.sciencedirect.com/science/article/pii/S0924271619301352

[101] Y. Wang, L. Gu, X. Li, R. Ren, Building extraction in multitemporal high-resolution remote sensing imagery using a multifeature lstm network, IEEE Geoscience and Remote Sensing Letters.

[102] S. Sun, L. Mu, L. Wang, P. Liu, X. Liu, Y. Zhang, Semantic segmentation for buildings of large intra-class variation in remote sensing images with o-gan, Remote Sensing 13 (3) (2021) 475.

[103] S. Touzani, J. Granderson, Open data and deep semantic segmentation for automated extraction of building footprints, Remote Sensing 13 (13). doi:10.3390/rs13132578.
URL https://www.mdpi.com/2072-4292/13/13/2578

[104] N. Yang, H. Tang, Geoboost: An incremental deep learning approach toward global mapping of buildings from vhr remote sensing images, Remote Sensing 12 (11) (2020) 1794.

[105] G. Sun, H. Huang, A. Zhang, F. Li, H. Zhao, H. Fu, Fusion of multiscale convolutional neural networks for building extraction in very high-resolution images, Remote Sensing 11 (3). doi:10.3390/rs11030227.
URL https://www.mdpi.com/2072-4292/11/3/227

[106] K. Song, J. Jiang, Agcdetnet: An attention-guided network for building change detection in high-resolution remote sensing images, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 4816–4831.

[107] R. Jaturapitpornchai, P. Rattanasuwan, M. Matsuoka, R. Nakamura, Corn: An alternative way to utilize time-series data of sar images in newly built construction detection, Remote Sensing 12 (6). doi:10.3390/rs12060990.
URL https://www.mdpi.com/2072-4292/12/6/990

[108] Z. Zheng, Y. Wan, Y. Zhang, S. Xiang, D. Peng, B. Zhang, Clnet: Cross-layer convolutional neural network for change detection in optical remote sensing imagery, ISPRS Journal of Photogrammetry and Remote Sensing 175 (2021) 247–267.

[109] L. Zhang, X. Hu, M. Zhang, Z. Shu, H. Zhou, Object-level change detection with a dual correlation attention-guided detector, ISPRS Journal of Photogrammetry and Remote Sensing 177 (2021) 147–160.

[110] Y. Gong, F. Zhang, X. Jia, X. Huang, D. Li, Z. Mao, Deep neural networks for quantitative damage evaluation of building losses using aerial oblique images: Case study on the great wall (china), Remote Sensing 13 (7) (2021) 1321.

[111] G. Abdi, S. Jabari, A multi-feature fusion using deep transfer learning for earthquake building damage detection, Canadian Journal of Remote Sensing (2021) 1–16.

[112] W. Yang, X. Zhang, P. Luo, Transferability of convolutional neural network models for identifying damaged buildings due to earthquake, Remote Sensing 13 (3) (2021) 504.

[113] Y. Cai, H. He, K. Yang, S. N. Fatholahi, L. Ma, L. Xu, J. Li, A comparative study of deep learning approaches to rooftop detection in aerial images, Canadian Journal of Remote Sensing (2021) 1–19.

[114] Z. Shu, X. Hu, J. Sun, Center-point-guided proposal generation for detection of small and dense buildings in aerial imagery, IEEE Geoscience and Remote Sensing Letters 15 (7) (2018) 1100–1104.

[115] Y. Liu, Z. Zhang, R. Zhong, D. Chen, Y. Ke, J. Peethambaran, C. Chen, L. Sun, Multilevel building detection framework in remote sensing images based on convolutional neural networks, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (10) (2018) 3688–3700. doi:10.1109/JSTARS.2018.2866284.

[116] F. Bi, J. Zhang, F. Pang, M. Bian, Y. Wang, Suburban building detection from optical remote sensing images based on a deformation adaptability model, Journal of the Indian Society of Remote Sensing 48 (6) (2020) 831–839.

[117] F. Alidoost, H. Arefi, A cnn-based approach for automatic building detection and recognition of roof types using a single aerial image, PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science 86 (5) (2018) 235–248.

[118] D. Chakraborty, S. Chowdhury, Identifying and counting of buildings using artificial neural network and reduced representation in high-resolution images, Geocarto International (2021) 1–16.

[119] M. Turgeon-Pelchat, S. Foucher, Y. Bouroubi, Deep learning-based classification of large-scale airborne lidar point cloud, Canadian Journal of Remote Sensing (2021) 1–15.

[120] X. Zhuo, F. Fraundorfer, F. Kurz, P. Reinartz, Optimization of openstreetmap building footprints based on semantic information of oblique uav images, Remote Sensing 10 (4) (2018) 624.

[121] L. Zhang, Z. Li, A. Li, F. Liu, Large-scale urban point cloud labeling and reconstruction, ISPRS Journal of Photogrammetry and Remote Sensing 138 (2018) 86–100. `doi:https://doi.org/10.1016/j.isprsjprs.2018.02.008`.
URL `https://www.sciencedirect.com/science/article/pii/S0924271618300376`

[122] D. Yu, S. Ji, J. Liu, S. Wei, Automatic 3d building reconstruction from multi-view aerial images with deep learning, ISPRS Journal of Photogrammetry and Remote Sensing 171 (2021) 155–170.

[123] F. Alidoost, H. Arefi, F. Tombari, 2d image-to-3d model: Knowledge-based 3d building reconstruction (3dbr) using single aerial images and convolutional neural networks (cnns), Remote Sensing 11 (19). `doi:10.3390/rs11192219`.
URL `https://www.mdpi.com/2072-4292/11/19/2219`

[124] D. Li, X. Shen, Y. Yu, H. Guan, J. Li, G. Zhang, D. Li, Building extraction from airborne multi-spectral lidar point clouds based on graph geometric moments convolutional neural networks, Remote Sensing 12 (19) (2020) 3186.

[125] L. Zhang, L. Zhang, Deep learning-based classification and reconstruction of residential scenes from large-scale point clouds, IEEE Transactions on Geoscience and Remote Sensing 56 (4) (2018) 1887–1897. `doi:10.1109/TGRS.2017.2769120`.

[126] C. Gevaert, C. Persello, F. Nex, G. Vosselman, A deep learning approach to dtm extraction from imagery using rule-based training labels, ISPRS Journal of Photogrammetry and Remote Sensing 142 (2018) 106–123. doi:https://doi.org/10.1016/j.isprsjprs.2018.06.001.
URL https://www.sciencedirect.com/science/article/pii/S0924271618301643

[127] Y. Cao, X. Huang, A deep learning method for building height estimation using high-resolution multi-view imagery over urban areas: A case study of 42 chinese cities, Remote Sensing of Environment 264 (2021) 112590.

[128] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Sparse generative adversarial network, Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019 (2019) 3063–3071arXiv:1908.08930, doi:10.1109/ICCVW.2019.00369.

[129] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.
URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-lear
pdf

[130] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, software available from tensorflow.org (2015).
URL https://www.tensorflow.org/

[131] Y. Long, G. S. Xia, S. Li, W. Yang, M. Y. Yang, X. X. Zhu, L. Zhang, D. Li, DiRS: On creating

benchmark datasets for remote sensing image interpretation, arXiv (2020) 1–22arXiv:2006.12485.

[132] V. Mnih, Machine Learning for Aerial Image Labeling, PhD Thesis (2013) 109.

[133] E. Maggiori, Y. Tarabalka, G. Charpiat, P. Alliez, Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark, International Geoscience and Remote Sensing Symposium (IGARSS) 2017-July (2017) 3226–3229. doi:10.1109/IGARSS.2017.8127684.

[134] ISPRS, 2d semantic labeling contest (2012).
URL http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html

[135] A. Song, J. Choi, Fully Convolutional Networks with Multiscale 3D Filters and Transfer Learning for Change Detection in High Spatial Resolution Satellite Images, Remote Sensing 12 (5) (2020) 799. doi:10.3390/rs12050799.
URL https://www.mdpi.com/2072-4292/12/5/799

[136] T. S. Catalog, Spacenet on amazon web services (aws) (2018).
URL https://spacenet.ai/datasets/

[137] A. Van Etten, D. Lindenbaum, T. Bacastow, SpaceNet: A remote sensing dataset and challenge series, arXivarXiv:1807.01232.

[138] A. Crowd, Ai crowd mapping challenge (2018).
URL https://www.aicrowd.com/challenges/mapping-challenge

[139] G. Labs, Open cities ai challenge dataset (2020).
URL https://doi.org/10.34911/rdnt.f94cxb

[140] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 9351, 2015, pp. 234–241. arXiv:1505.04597, doi:10.1007/978-3-319-24574-4_28.

[141] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings - 30th IEEE Conference on Computer Vision and Pattern

Recognition, CVPR 2017, Vol. 2017-Janua, 2017, pp. 936–944. arXiv:1612.03144, doi:10.1109/CVPR.2017.106.

[142] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation (2016). arXiv:1511.00561.

[143] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 801–818.

[144] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[145] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision (2015). arXiv:1512.00567.

[146] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks (2017). arXiv:1611.05431.

[147] J. Avbelj, R. Muller, R. Bamler, A metric for polygon comparison and building extraction evaluation, IEEE Geoscience and Remote Sensing Letters 12 (1) (2015) 170–174. doi:10.1109/LGRS.2014.2330695.

[148] D. H. Douglas, T. K. Peucker, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature, Cartographica: the international journal for geographic information and geovisualization 10 (2) (1973) 112–122.

[149] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, H. C. Wolf, Parametric correspondence and chamfer matching: Two new techniques for image matching, Tech. rep., SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGENCE CENTER (1977).

[150] H. Fan, H. Su, L. J. Guibas, A point set generation network for 3d object reconstruction from a single image, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 605–613.

[151] X. Sun, J. Wu, X. Zhang, Z. Zhang, C. Zhang, T. Xue, J. B. Tenenbaum, W. T. Freeman, Pix3d: Dataset and methods for single-image 3d shape modeling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2974–2983.

[152] L. Castrejón, K. Kundu, R. Urtasun, S. Fidler, Annotating object instances with a polygon-rnn, in: CVPR, 2017.

[153] D. Cheng, R. Liao, S. Fidler, R. Urtasun, Darnet: Deep active ray network for building segmentation (2019). `arXiv:1905.05889`.

[154] D. Acuna, H. Ling, A. Kar, S. Fidler, Efficient interactive annotation of segmentation datasets with polygon-rnn++.

[155] H. Ling, J. Gao, A. Kar, W. Chen, S. Fidler, Fast interactive object annotation with curve-gcn, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5257–5266.

[156] F. Zhang, N. Nauata, Y. Furukawa, Conv-mpn: Convolutional message passing neural network for structured outdoor architecture reconstruction (2021). `arXiv:1912.01756`.

[157] T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks (2017). `arXiv:1609.02907`.

[158] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, P. Vandergheynst, Geometric deep learning: going beyond euclidean data, IEEE Signal Processing Magazine 34 (4) (2017) 18–42.

[159] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, Y.-G. Jiang, Pixel2mesh: Generating 3d mesh models from single rgb images, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 52–67.

[160] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, G. E. Dahl, Neural message passing for quantum chemistry, ArXiv abs/1704.01212.

[161] Z. Li, J. D. Wegner, A. Lucchi, Topological map extraction from overhead images (2019). `arXiv:1812.01497`.

[162] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.

[163] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8759–8768.

[164] S. Peng, W. Jiang, H. Pi, X. Li, H. Bao, X. Zhou, Deep snake for real-time instance segmentation, in: CVPR, 2020.

[165] Y. Cao, J. Xu, S. Lin, F. Wei, H. Hu, Gcnet: Non-local networks meet squeeze-excitation networks and beyond (2019). arXiv:1904.11492.

[166] J. Hu, L. Shen, S. Albanie, G. Sun, E. Wu, Squeeze-and-Excitation Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (8) (2020) 2011–2023. arXiv:1709.01507, doi:10.1109/TPAMI.2019.2913372.

[167] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, Cbam: Convolutional block attention module (2018). arXiv:1807.06521.

[168] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, B. Xiao, Deep High-Resolution Representation Learning for Visual Recognition (2019) 1–17arXiv:1908.07919.
URL http://arxiv.org/abs/1908.07919

[169] Y. Yuan, X. Chen, J. Wang, Object-Contextual Representations for Semantic SegmentationarXiv:1909.11065.
URL http://arxiv.org/abs/1909.11065

[170] Facebook, Papers with code (2018).
URL https://www.paperswithcode.com

[171] Y. Yuan, X. Chen, X. Chen, J. Wang, Segmentation transformer: Object-contextual representations for semantic segmentation (2021). arXiv:1909.11065.

[172] Q. Chen, L. Wang, Y. Wu, G. Wu, Z. Guo, S. L. Waslander, Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings, ISPRS Journal of Photogrammetry and Remote Sensing 147 (November 2018) (2019) 42–55. doi:10.1016/j.isprsjprs.2018.11.011.
URL https://doi.org/10.1016/j.isprsjprs.2018.11.011

[173] P. Borba, F. de Carvalho Diniz, N. C. da Silva, E. de Souza Bias, Building footprint extraction using deep learning semantic segmentation techniques: Experiments and results, in: 2021

IEEE International Geoscience and Remote Sensing Symposium IGARSS, 2021, pp. 4708–4711. doi:10.1109/IGARSS47720.2021.9553855.

[174] P. Borba, phborba/DeepLearningTools: First release (oct 2020). doi:10.5281/zenodo.4140680.
URL https://doi.org/10.5281/zenodo.4140680

[175] I. Loshchilov, F. Hutter, Decoupled weight decay regularization (2019). arXiv:1711.05101.

[176] L. N. Smith, A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay (2018). arXiv:1803.09820.

[177] L. N. Smith, Cyclical learning rates for training neural networks, Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017 (April 2015) (2017) 464–472. arXiv:1506.01186, doi:10.1109/WACV.2017.58.

[178] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826.

[179] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, 2017.

[180] J. Zhang, T. He, S. Sra, A. Jadbabaie, Why gradient clipping accelerates training: A theoretical justification for adaptivity (2020). arXiv:1905.11881.

[181] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, A. G. Wilson, Averaging weights leads to wider optima and better generalization (2019). arXiv:1803.05407.

[182] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, H. Wu, Mixed precision training (2018). arXiv:1710.03740.

[183] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034.

[184] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, Journal of Machine Learning Research 9 (2010) 249–256.

[185] E. M. A. Xavier, F. J. Ariza-L, Opez, M. A. Ure˜na, F. J. Ariza-López, M. A. Ureña, A Survey of Measures and Methods for Matching Geospatial Vector Datasets, ACM Comput. Surv 49. `doi:10.1145/2963147`.
URL `http://dx.doi.org/10.1145/2963147`

[186] T. Devogele, J. Trevisan, L. Raynal, Building a multi-scale database with scale-transition relationships, in: International symposium on spatial data handling, Citeseer, 1996, pp. 337–351.

[187] W. Rucklidge, Efficient visual recognition using the Hausdorff distance, Springer-Verlag, 1996.

[188] T. Devogele, A new merging process for data integration based on the discrete fréchet distance, in: Advances in spatial data handling, Springer, 2002, pp. 167–181.

[189] A. Kolmogorov, Sulla determinazione empirica di una lgge di distribuzione, Inst. Ital. Attuari, Giorn. 4 (1933) 83–91.

[190] T. W. Anderson, D. A. Darling, Asymptotic Theory of Certain ”Goodness of Fit” Criteria Based on Stochastic Processes, The Annals of Mathematical Statistics 23 (2) (1952) 193 – 212. `doi:10.1214/aoms/1177729437`.
URL `https://doi.org/10.1214/aoms/1177729437`

[191] M. G. Kendall, A new measure of rank correlation, Biometrika 30 (1/2) (1938) 81–93.

[192] G. W. Corder, D. I. Foreman, Nonparametric statistics for non-statisticians (2011).

[193] A. R. Gilpin, Table for conversion of kendall's tau to spearman's rho within the context of measures of magnitude of effect for meta-analysis, Educational and Psychological Measurement 53 (1) (1993) 87–92. `arXiv:https://doi.org/10.1177/0013164493053001007`, `doi:10.1177/0013164493053001007`.
URL `https://doi.org/10.1177/0013164493053001007`

[194] H. Liu, F. Liu, X. Fan, D. Huang, Polarized self-attention: Towards high-quality pixel-wise regression (2021). `arXiv:2107.00782`.

[195] F. C. Akyon, C. Cengiz, S. O. Altinuc, D. Cavusoglu, K. Sahin, O. Eryuksel, SAHI: A lightweight vision library for performing large scale object detection and instance segmenta-

tion (Nov. 2021). doi:10.5281/zenodo.5718950.
URL https://doi.org/10.5281/zenodo.5718950

[196] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, arXiv preprint arXiv:2103.14030.

6.2 Discussão do Artigo de Resultados

Tendo em vista as deficiências de métodos tradicionais para a extração de polígonos e subsequente generalização apontadas na introdução do capítulo 6, bem como as novas técnicas propostas por Girard et al. (2020) e por Girard et al. (2021), decidiu-se realizar uma pesquisa bibliográfica complementar ao que foi realizado no capítulo 4.

Logo, foi realizada uma meta-análise utilizando a base de dados Web of Science, com as palavras chaves ”*building extraction*” e o intervalo de tempo dos últimos cinco anos. Essa pesquisa teve por objetivo identificar aspectos primordiais relacionados ao tema de extração dos polígonos dos prédios (*building footprint extraction*), a saber: principais trabalhos, problemas de pesquisa preponderantes, os mais relevantes conjuntos de dados, os *frameworks* utilizados para o treinamento das redes neurais, os formatos dos dados de entrada e saída, os tipos de insumos, as funções de perda utilizadas, as técnicas de *data augmentation*, os trabalhos com código disponível, as arquiteturas e *backbones* utilizados e as métricas de avaliação utilizadas.

Em adição, após aplicar diversos filtros que visavam retirar trabalhos que não eram relevantes para a pesquisa, tal análise partiu de 99 artigos e identificou que o principal problema de pesquisa encontrado na busca é a extração do formato de prédios. Ademais, o *framework* mais popular dentre essas aplicações foi o PyTorch, os *datasets* mais utilizados foram o *WHU Aerial Building dataset*, o *Massachusetts Building dataset*, e o *INRIA Aerial Labeling dataset*. Além disso, constatou-se que poucos autores disponibilizaram o código na internet, apenas 13 dos 99 artigos analisados. Também constatou-se que a grande maioria dos artigos apenas extrai os dados no formato *raster*. Apenas 7 dos 99 artigos estudados abordam extração de vetores.

Em particular, focou-se nos 7 artigos que versam sobre métodos de obtenção de dados no formato vetorial. Dos trabalhos analisados, escolheu-se apenas um, de título ”*Building outline delineation: From aerial images to polygons with an improved end-to-end learning network*” (ZHAO et al., 2021). Tal escolha foi feita em virtude dos resultados apresentados e da arquitetura proposta, denominada ModPolyMapper no contexto dessa pesquisa.

Outrossim, o ModPolyMapper (ZHAO et al., 2021) é basicamente uma rede de detecção de objetos (*object detection*), acoplada a uma Recursive Neural Network (RNN). O ramo que detecta objetos é uma Mask R-CNN (HE et al., 2017) de backbone formado pela combinação de uma Feature Pyramid Network (FPN) com a ResNet-101. A RNN se acopla à saída da operação de RoiAlign, que ocorre no ramo de localização da rede. A técnica proposta é uma evolução da PolyMapper (LI et al., 2019), que difere da ModPolyMapper basicamente no *backbone*, uma VGG-16 (SIMONYAN; ZISSERMAN, 2015).

Ademais, foram pesquisados métodos semelhantes em publicações de Ciência da Computação. Como resultado dessa busca, pode-se citar os métodos PolygonRNN (CASTREJÓN et al., 2017), PolygonRNN++ (ACUNA et al., 2018), DARNet (CHENG et al., 2019), Curve-GCN (LING et al., 2019), e Conv-MPN (ZHANG et al., 2021). Tais métodos partilham da mesma deficiência: como foram concebidos para serem utilizados em ferramentas de rotulação semi-supervisionada de dados, eles dependem que a imagem de entrada contenha apenas um objeto a ser delimitado.

Logo, conforme discutido no artigo, os métodos supramencionados não são ideais para a produção cartográfica, podendo porém ser utilizados em ferramentas de extração semi-supervisionadas. Vale salientar que todas as técnicas provenientes de publicações de Ciência da Computação fornecem os códigos *online*, mostrando uma cultura de compartilhamento de dados diferente do que foi observado nas publicações de Sensoriamento Remoto consideradas: apenas 13% dos artigos estudados disponibilizaram os códigos na internet.

Em seguida, procedeu-se às implementações. Devido aos diversos códigos disponíveis estarem escritos utilizando o PyTorch, dada a popularidade identificada na pesquisa bibliográfica do terceiro artigo, e devido à dificuldade encontrada em realizar algumas atividades de *debug* durante o desenvolvimento do *framework* baseado no TensorFlow, decidiu-se migrar a pesquisa para o PyTorch.

Logo, foi desenvolvido um *framework* baseado em software livre e de código aberto, com licença GPL-2.0. A solução, denominada pytorch_segmentation_models_trainer (BORBA, 2021), é desenvolvida utilizando a linguagem de programação Python e se baseia nas bibliotecas PyTorch, PyTorch Lightning, Hydra (YADAN, 2019) e segmentation_models.pytorch.

Particularmente, grande parte dos modelos utilizados são da segmentation_models.pytorch, com exceção de alguns que foram implementados no escopo da pesquisa. Além disso, utiliza-se o PyTorch Lightning para simplificar e acelerar os processos de treinamento, especialmente por meio das funcionalidades de treinamento distribuído, *gradient clipping*, *stochastic weight averaging*, *mixed precision* (MP) e *gradient scaling*, que é necessária quando se utiliza MP.

O *framework* também utiliza o pacote Hydra (YADAN, 2019) para armazenar as configurações dos treinamentos, inferências, construções de *dataset* e avaliações por meio de arquivos no formato yaml. Para as rotinas de *data augmentation*, foram utilizadas as bibliotecas albumentations e kornia. Esta última é interessante pois propicia que operações de data augmentation sejam realizadas na GPU, melhorando consideravelmente o desempenho dos algoritmos, dessa forma reduzindo o tempo de treinamento.

Vale salientar que todas as tecnologias utilizadas foram encontradas na ocasião da elaboração do primeiro artigo. Todos os experimentos foram realizados utilizando a biblioteca desenvolvida, a qual está disponível no gerenciador de pacotes pip[7] do Python e no repositório do GitHub[8].

Não obstante, como o código da ModPolyMapper não está disponível online, entrou-se em contato com os autores por e-mail solicitando acesso aos códigos e autorização para adaptá-lo para o *framework open-source* da pesquisa. Os autores do artigo forneceram as implementações e não se opuseram à adaptação. Destaca-se que os códigos fornecidos pelos autores foram implementados em Python utilizando a biblioteca Tensorflow. Logo, o modo como o problema foi resolvido foi entendido e, em seguida, uma versão foi desenvolvida utilizando o PyTorch, que está disponível no pytorch_segmentation_models_trainer.

Para os experimentos, decidiu-se comparar a ResNet-101 U-Net Frame Field com a ModPolyMapper. Entretanto, resolveu-se também experimentar variações no método *Frame Field*, utilizando algumas das arquiteturas encontradas nas pesquisas bibliográficas para os artigos 1 e 2. Logo, realizou-se experimentos utilizando a HRNet e a DeepLabV3+ como arquiteturas alternativas combinadas ao *Frame Field*. Como na época a HRNet não estava integrada à biblioteca segmentation_models.pytorch, ela foi implementada no *framework*, baseando-se em códigos disponível online, com licenças compatíveis com a GPL-2.0

Similarmente, foram realizados alguns testes preliminares com a SE-ResNeXt-101 + U-Net, encontrada no segundo artigo, combinados ao *Frame Field*. Contudo, tal solução foi abandonada devido ao uso elevado de memória. Além disso, os resultados do estado da arte de segmentação semântica no site *Papers With Code* (FACEBOOK, 2018) apontava para a HRNet como melhor arquitetura nas competições do *dataset* Cityscapes. Esse fato foi comprovado pelos treinamentos preliminares que realizamos antes de conduzir os experimentos sob as condições escolhidas no estudo.

Em sequência, foram conduzidos os experimentos supramencionados, utilizando o *Brazilian Army Geographic Service Building dataset*. Além desse conjunto de dados, também foram realizados experimentos utilizando o AICrowd, uma vez que este foi utilizado em ambos os artigos selecionados. Todos os parâmetros dos experimentos foram descritos no artigo da subseção 6.1, porém, um aspecto que vale ser ressaltado é o uso de técnicas para melhorar o treinamento, como *gradient clipping* e *stochastic weight averaging*, estudadas no primeiro artigo (subseção 4.1). Tais procedimentos proporcionaram melhores resultados, sendo necessário menos épocas para obter pro-

[7] https://pypi.org/project/pytorch-segmentation-models-trainer/

[8] https://github.com/phborba/pytorch_segmentation_models_trainer

dutos aceitáveis.

Em adição, foram escolhidas as mesmas métricas de avaliação dos artigos selecionados: Polygons and Line Segments (PoLis) (AVBELJ et al., 2015), utilizada por Zhao et al. (2021), e a Mean Max Tangent Angle Errors (MMTAE), proposta por Girard et al. (2021). Também foi utilizada a Intersection over Union (IoU) com os vetores obtidos e foi medido as omissões e comissões, medidas que são importantes quando se analisam os produtos sob a óptica da ET-CQDG.

Como resultado dos treinamentos, o melhor método aplicado ao *dataset* AICrowd foi o proposto pelos autores dessa pesquisa, o HRNet OCR W48 Frame Field, o qual atingiu 1,7 na PoLiS, 43,89 na MMTAE, 0,86 no IoU, 12% de omissões e 12% de excessos. Exceto pela métrica MMTAE, o ModPolyMapper foi o pior dentre as técnicas consideradas. No entanto, o mesmo não ocorreu no *BAGS building dataset*, uma vez que o ModPolyMapper obteve melhores respostas em todas as métricas consideradas (1,75 de PoLiS, 7,55 de MMTAE, 0,78 de IoU, 1% de omissões e 1% de excessos) e o HRNet OCR W48 Frame Field as piores (2,25 de PoLiS, 38,72 de MMTAE, 0,62 de IoU, 11% de omissões e 11% de excessos).

Logo, foram realizadas inspeções visuais nos resultados obtidos com a finalidade de tentar entender o porquê da discrepância entre os experimentos. Em função dessa diferença, suspeitou-se que podia haver um comportamento diferente, baseado na densidade de edificações das amostras dos conjuntos de dados. Logo, formulou-se uma hipótese de que o método HRNet OCR W48 Frame Field pudesse funcionar melhor em regiões mais densas e, em contraste, o ModPolyMapper pudesse ter melhor desempenho em regiões esparsas.

Sendo assim, partiu-se para uma investigação estatística para tentar comprovar ou refutar a conjectura proposta. Calculou-se as seguintes quantidades por amostra: densidade de edificações, os valores médios de PoLiS e as taxas de omissão e comissão. Em seguida, realizou-se testes de normalidade utilizando o teste de uma amostra de Kolmogorov-Smirnov (KOLMOGOROV, 1933), com um intervalo de confiança de 95% para as variáveis densidade de edificações e PoLiS média. Desse teste constatou-se que ambas as variáveis não seguem uma distribuição normal.

Para as taxas de omissão e comissão, testou-se a normalidade dos dados por meio do teste de Anderson-Darling (ANDERSON; DARLING, 1952), uma vez que há repetição de valores nos dados. O intervalo de confiança adotado também foi de 95% e também concluiu-se que as variáveis não seguiam uma distribuição normal.

Como todas as variáveis não seguem uma distribuição normal, foi realizado um teste de correlação de Kendall (KENDALL, 1938) e deduziu-se que há uma correlação moderada entre a densidade dos tiles e a métrica PoLiS. Porém, constatou-se apenas uma correlação fraca entre as taxas de omissão e comissão com a densidade dos tiles. Logo, concluiu-se que a discrepância nos resultados pode ser explicada em partes pelas diferentes densidades nas amostras dos treinamentos.

6.3 Discussão dos Resultados à Luz das Normas do Sistema Cartográfico Nacional (SCN)

No contexto do Sistema Cartográfico Nacional (SCN), a ET-CQDG define os valores de excesso e omissão para pequenas escalas na tabela 30 da norma, cujo extrato pode ser visualizado na tabela 1.

Tabela 1: Valores de excessos e omissões para Conjunto de Dados Geoespaciais Vetoriais (CDGV) em pequenas escalas, segundo a tabela 30 da ET-CQDG (DSG, 2016a).

Linha	Escopo	Elemento	Medida	Parâmetro	Procedimento	Resultado
2	Produto	Excesso	102	-	Direto interno. Inspeção completa	Conformidade M < 1%
3	Produto	Omissão	103	-	Direto interno. Amostragem	Conformidade M < 4%

Vale salientar que as avaliações dos resultados foram procedimentos diretos internos de inspeção completa. A medida 103, mostrada na tabela 1, sugere procedimento por amostragem, porém, como todos os polígonos da verdade de campo do conjunto de dados de teste podem ser utilizados, foi escolhido aplicar um mecanismo de controle de qualidade mais rigoroso que o previsto na norma, por inspeção completa, como o previsto na medida 102. Dos resultados apresentados na subseção 6.1 para o *BAGS building dataset*, pode-se perceber que as taxas de excesso e omissão já atendem ao previsto na norma.

Entretanto, o melhor método no dataset AICrowd não seria aprovado em nenhuma das medidas supralistadas, dado que a técnica HRNet OCR W48 Frame Field resultou em 12% de omissões e 12% de excessos.

Porém, até o presente momento, ainda não foi realizada a verificação da qualidade posicional dos dados extraídos. Para que se possa realizar essa aferição, deve-se considerar os aspectos da escala: segundo a tabela 1 da ET-ADGV, a área mínima de uma edificação $1mm^2$ na escala da carta. Como a hipótese de pesquisa define a escala de trabalho como 1:25.000, o valor que deve ser adotado para filtrar as informações é de $625m^2$, ou seja, polígonos com área inferior ao limiar devem ser representados como ponto e aqueles com área superior são mantidos intactos.

Logo, como análise complementar que não foi abordada no artigo, foi calculado o Erro Médio (EM) e o Erro Padrão (EP) para que se possa determinar o Padrão de Exatidão Cartográfico (PEC) planimétrico, definidos na tabela 31 da ET-CQDG, ilustrado na tabela 2.

Tabela 2: Valores de erro médio (EM) e erro padrão (EP), em metros na planimetria, para Conjunto de Dados Geoespaciais Vetoriais (CDGV) em pequenas escalas, na escala 1:25.000, segundo a tabela 31 da ET-CQDG (DSG, 2016a).

PEC	1:25.000	
	EM	EP
A	7,00	4,25
B	12,50	7,50
C	20,00	12,50
D	25,00	15,00

Sendo assim, foi adotado o seguinte procedimento para a aferição da qualidade planimétrica: para um dado polígono da verdade de campo, se ele tem área menor que $625m^2$, deve-se transformá-lo e o polígono predito correspondente para ponto e calcular o deslocamento horizontal entre eles. Tal deslocamento entrará no cálculo do EM e do EP.

Para os casos de polígonos representáveis em escala, considerou-se os polígonos pareados por meio do maior valor da razão interseção por união (IoU) e, em seguida, para cada par de geometrias, foi realizado um pareamento de vértices por vizinho mais próximo.

A seguir, calculou-se a distância entre os pontos pareados. Essa medida é o erro do vértice considerado, dado que um dos polígonos corresponde à verdade de campo e o outro ao valor predito. O processo para o cálculo do PEC pode ser visualizado na figura 6. Os resultados dessa nova análise podem ser visualizados na tabela 3.

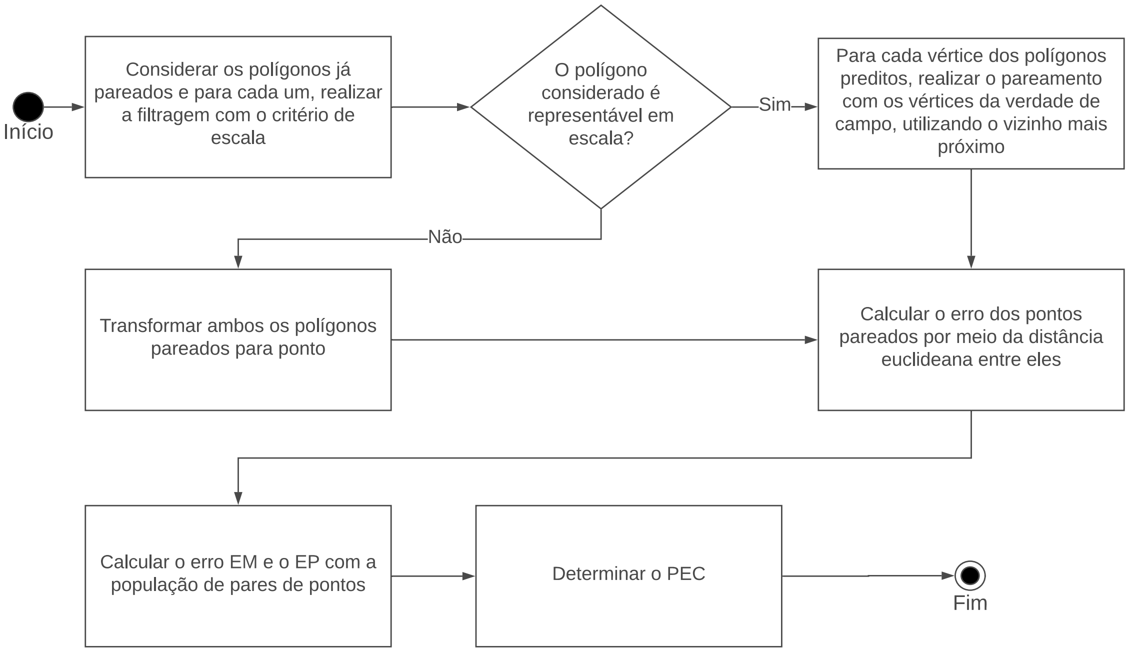

Início
Considerar os polígonos já pareados e para cada um, realizar a filtragem com o critério de escala
O polígono considerado é representável em escala?
Sim
Para cada vértice dos polígonos preditos, realizar o pareamento com os vértices da verdade de campo, utilizando o vizinho mais próximo
Não
Transformar ambos os polígonos pareados para ponto
Calcular o erro dos pontos pareados por meio da distância euclideana entre eles
Calcular o erro EM e o EP com a população de pares de pontos
Determinar o PEC
Fim

Figura 6: Fluxograma do cálculo do PEC.

Tabela 3: Valores dos Erros Médios (EM), Erros Padrão (EP) e do Padrão de Exatidão Cartográfico (PEC) para a escala 1:25.000, calculados para todos os CDGV dos experimentos realizados.

Dataset	Método	EM ↓	EP ↓	PEC
AICrowd	HRNet OCR W48 Frame Field	1.44	**2.41**	A
	ResNet-101 UNet Frame Field	1.35	2.70	A
	ResNet-101 DeepLabV3+ Frame Field	1.52	2.55	A
	ModPolyMapper	**1.41**	3.02	A
BAGS	ModPolyMapper	1.15	1.85	A
	ResNet-101 UNet Frame Field	**0.70**	**0.84**	A
	ResNet-101 DeepLabV3+ Frame Field	0.72	0.98	A
	HRNet OCR W48 Frame Field	0.98	1.01	A

Dos dados apresentados na tabela 3, pode-se observar que todos os resultados dos experimentos foram classificados como PEC A, ou seja, são dados com a melhor classificação de acurácia posicional dentro dos padrões do SCN.

Portanto, conclui-se que todos os métodos analisados atingiram acurácia posicional adequada para a produção de cartas topográficas na escala 1:25.000 em ambos os conjuntos de dados. Além disso, considerando o *BAGS Buildings*, conforme mencionado anteriormente, somente o método ModPolyMapper atingiu a taxa de omissões e excessos aceitáveis pela ET-CQDG, enquanto todos as outras técnicas nesse *dataset* foram reprovados em pelo menos um desses quesitos. Em contrapartida, no AICrowd, todas as técnicas consideradas foram reprovadas em ambas medidas de qualidade.

Entretanto, vale ressaltar que no processo de produção cartográfica, há diversas revisões, as quais visam, entre outros aspectos, zerar os erros de omissão e excesso, na medida do possível e dentro dos parâmetros da ET-CQDG. Como é possível extrair um grande volume de dados ao utilizar as técnicas estudadas, considera-se que o uso de tais metodologias em processos de produção cartográfica pode acarretar um ganho produtivo considerável.

7 Conclusões e Recomendações

A presente pesquisa estudou técnicas de Inteligência Artificial, especificamente técnicas de *Deep Learning* baseadas em redes neurais convolucionais visando extrair as geometrias de edificações em imagens de altíssima resolução. O estudo abordou desde conhecimentos básicos, aos métodos do estado da arte.

Adicionalmente, a dissertação foi dividida em três artigos, nos quais procurou-se estudar sistematicamente as ideias e métodos necessários para que os objetivos do trabalho pudessem ser alcançados pudesse, e que pudesse validar ou refutar a hipótese de pesquisa: **é possível treinar uma rede convolucional profunda que seja capaz de segmentar e extrair feições vetoriais no formato poligonal de edificações, geometricamente consistentes, com acurácia compatível com a escala 1:25.000, em conformidade com as normas definidas pelo SCN**.

Primeiramente, foi elaborado um artigo científico (capítulo 4) para realizar a revisão bibliográfica de todo o arcabouço teórico necessário para o profundo entendimento da área. Além disso, dentre outros conhecimentos, foram identificados métodos do estado da arte, técnicas, arquiteturas, *backbones*, tecnologias e oportunidades de pesquisa relativos ao campo de Segmentação Semântica utilizando *Deep Learning*.

Em seguida, foram conduzidos uma série de experimentos para testar alguns dos conhecimentos identificados no primeiro artigo. Também foi construído um conjunto de dados denominado *Brazilian Army Geographic Service Building Dataset (BAGS Buildings dataset)*, com cujas imagens os testes foram realizados.

Além disso, foram escritas na linguagem de programação Python duas soluções baseadas em software livre e de código aberto: um complemento para o QGIS para construção e manipulação de *datasets* e um *framework* baseado no Tensorflow para o treinamento de redes neurais convolucionais de segmentação semântica.

Os resultados dos experimentos supramencionados foram consolidados no segundo artigo da pesquisa (capítulo 5) e se pode identificar que dentre os testes, a melhor combinação de métodos foi a SE-ResNeXt-101+U-Net, com os seguintes valores para as métricas escolhidas no trabalho: 0,451 de IoU, 0,518 de *Precision*, 0,774 de *Recall* e 0,621 de índice F1. Visualmente, foram identificados vários problemas com omissões e artefatos nas máscaras segmentadas.

Entretanto, como a presente dissertação tem por finalidade estudar formas de extração de polígonos no formato vetorial, fazia-se necessário que uma pesquisa bibliográfica focada em formas de se extrair polígonos fosse realizada. Ademais, tal busca foi realizada de forma específica em publicações de Sensoriamento Remoto, considerando o tópico de extração de edificações utilizando técnicas baseadas em *Deep*

Learning, apresentado no capítulo 6.

No levantamento supramencionado, foram identificados aspectos como as 10 publicações mais citadas, os temas de pesquisa, a disponibilidade do código da pesquisa, os conjuntos de dados utilizados, os tipos de imagens utilizadas nos experimentos, os tipos de saída. Particularmente, foi dado um enfoque aos trabalhos que tratavam o vetor como saída. Foi feita também uma pesquisa complementar, em periódicos de ciência da computação, nos quais foram encontrados outros métodos. Após análise, foram escolhidas duas técnicas para os testes: a Polygonization by Frame Field (GIRARD et al., 2021) e a ModPolyMapper (ZHAO et al., 2021).

Além disso, foram propostas variações ao método de Girard et al. (2021), utilizando arquiteturas como a HRNet e a DeepLabV3+. Os experimentos foram realizados utilizando os dados do *BAGS Buildings dataset* e do AICrowd, este também utilizado por (ZHAO et al., 2021; GIRARD et al., 2021). Em adição, todos os métodos foram implementados em uma solução baseada em *software* livre e de código aberto, com licença GPL 2.0. Tal biblioteca desenvolvida é baseada, dentre outras tecnologias, em PyTorch.

Outrossim, foram realizados 8 experimentos e concluiu-se que no AICrowd, o melhor resultado foi da HRNet OCR W48 Frame Field, método proposto pelos autores, atingindo valores de 1,7 para a PoLiS, MMTAE de 43,89, IoU de 0,86, 12% de omissões e 12% de excessos.

Contudo, no *BAGS Buildings dataset*, os melhores valores das métricas consideradas foram obtidos pelo ModPolyMapper: 1,75 na PoLiS, 7,55 de MMTAE, 0,78 de IoU, 1% de omissões e 1% de excessos. Logo, após as análises descritas no artigo, concluiu-se que o método HRNet OCR W48 Frame Field funcionou melhor em áreas densamente edificadas, enquanto a ModPolyMapper em regiões esparsas.

Adicionalmente, foram apresentadas análises de qualidade dos dados obtidos nos experimentos da última publicação à luz das normas do Sistema Cartográfico Nacional (SCN). Foram realizadas considerações sobre a porcentagem de excessos e omissões (medidas 102 e 103 da ET-CQDG), além do cálculo do PEC na escala 1:25.000.

Quanto aos excessos e omissões, no conjunto de dados *BAGS Buildings dataset*, a ModPolymapper apresentou taxas em conformidade com norma. Já no AICrowd, nenhum dos métodos foi aprovado nas duas medidas de qualidade em questão. Por outro lado, todos os experimentos foram considerados com PEC A, melhor classificação de acurácia posicional prevista na norma.

Sendo assim, considera-se que é possível utilizar tanto o ModPolyMapper, quanto o HRNet OCR W48 Frame Field em processos de produção cartográfica, visto que ambos obtiveram acurácia posicional aderentes à melhor classificação prevista na ET-

CQDG.

Além do mais, quanto às omissões e aos excessos, ambos podem ser sanados em processos de revisão. Como a aquisição manual de polígonos de edificação é um processo demorado, espera-se que o uso dos métodos levantados pode vir a trazer um ganho considerável produtivo.

Para fins de comparação, a construção do *BAGS Buildings dataset* demandou 5400 horas de trabalho dos operadores do 1º Centro de Geoinformação (1º CGEO) para a extração manual das 1,6 milhões de edificações, em uma área equivalente à 840 cartas topográficas na escala 1:25.000.

Já o processo automatizado proposto nessa dissertação, o treinamento de cada método com o servidor de *Machine Learning* do 1º CGEO demorou em média 5 dias. Ademais, para extrair os polígonos de uma área de 1.981.728 km^2, o equivalente à aproximadamente 16,3 cartas na escala 1:25.000 no sul do Brasil, foi necessário cerca de uma hora. Ou seja, para extrair as informações de uma área equivalente às 840 cartas, seria necessário aproximadamente 51,29 horas para a inferência, além das 120 horas do treinamento (5 dias).

Portanto, estima-se que o tempo total para extrair os polígonos utilizando os métodos automatizados seria de 171,23 horas, 31,5 vezes mais rápido que o processo manual.

Porém, é preciso frisar que, devido aos erros de omissão e excessos observados, etapas de revisão e correção já previstas na linha de produção da DSG são necessárias para que os produtos estejam em conformidade com os padrões do SCN.

Logo, como etapas de revisão e correção são essencialmente manuais, o tempo de produção estimado aumentará, no entanto, pode-se observar um ganho produtivo considerável ao se aplicar as técnicas propostas nessa dissertação.

Diante do exposto, considerando que os dados obtidos tem acurácia posicional PEC A para 1:25.000, que foi possível, sem revisões, em um dos métodos atingir níveis de omissão e excesso compatíveis com a ET-CQDG, que as taxas em questão nos outros métodos podem ser sanados em processos de revisão e correção, e que as técnicas propostas nessa dissertação são bem mais rápidas que o processo manual, conclui-se que a hipótese de pesquisa foi validada, ou seja, **é possível treinar uma rede convolucional profunda que seja capaz de segmentar e extrair feições vetoriais no formato poligonal de edificações, geometricamente consistentes, com acurácia compatível com a escala 1:25.000, em conformidade com as normas definidas pelo SCN**.

Por conseguinte, uma vez que a hipótese científica foi validada, o objetivo da pesquisa também foi atingido, dado que verificou-se que é possível treinar uma rede neural convolucional profunda, que seja capaz de segmentar imagens de satélite de altíssima resolução, com a finalidade de extrair geometrias de edificação com características e níveis de qualidades adentes aos padrões definidos pelo SCN, na escala 1:25.000.

Com relação aos objetivos específicos, considera-se que todos foram atingidos, uma vez que o presente estudo apresentou diversas arquiteturas que podem obter as geometrias de edificações, que foi determinado o melhor método tanto para áreas densamente edificadas, quanto para regiões esparsas. Além disso, conseguiu-se extrair os polígonos com qualidade aderente aos padrões definidos pelo SCN. De maneira detalhada, cada objetivo específico foi atingido da seguinte forma:

(a) Foram identificadas arquiteturas de redes neurais capazes de extrair informações para a obtenção de geometrias de edificação, como a HRNet w48 OCR Frame Field e a ModPolyMapper;

(b) Verificou-se que a melhor arquitetura para ser utilizada em atividade de mapeamento depende da cena: A HRNet OCR W48 Frame Field funciona melhor em cenas densamente edificadas, enquanto a ModPolyMapper apresenta melhores resultados em cenas esparsas;

(c) Identificou-se que o melhor método, ou combinação de métodos, foi por meio de processos ”*end-to-end*”, como os mencionados no item anterior. Além disso, o melhor método depende da natureza da cena;

(d) Foi desenvolvida uma metodologia de treinamento de rede neural convolucional profunda capaz de segmentar imagens de satélite de altíssima resolução com a finalidade de extrair geometrias de edificação com características e níveis de qualidades aderentes aos padrões definidos pelo SCN, na escala 1:25.000. Tal metodologia é materializada por meio dos arquivos de configuração utilizados para treinar os métodos descrito nesta pesquisa e pelos pesos das redes treinadas. Vale ressaltar que tais arquivos serão disponibilizados nos repositórios do GitHub dos *frameworks* desenvolvidos no contexto deste trabalho;

(e) Foram desenvolvidos dois *frameworks* em Python, um baseado no Tensorflow, denominado segmentation_models_trainer, e outro baseado no PyTorch, denominado pytorch_segmentation_models_trainer. Ambos os desenvolvimentos são baseadas em soluções livres e de código aberto. Além do código estar disponível para a comunidade científica, para os órgãos de mapeamento e para a sociedade civil em geral, reforça-se que as configurações supramencionadas e os pesos das redes treinadas serão disponibilizados no repositório do pytorch_segmentation_models_trainer ao final da pesquisa, para que novas pesquisas possam ser realizadas e que os resultados aqui propostos possam ser verificados; e

(f) Após avaliação, atestou-se que os polígonos obtidos ao final do processo de extração possuem acurácia compatível com os parâmetros definidos pelas normas do SCN, conforme apresentado na subseção 6.3.

Como contribuições da pesquisa, pode-se citar:

(a) Artigo de levantamento e análise do estado da arte de Segmentação Semântica em aplicações de sensoriamento remoto;

(b) Artigo de aplicações de técnicas de *deep learning* aplicadas ao problema de extração de prédios por meio de imagens de satélites;

(c) Artigo com os resultados da pesquisa;

(d) Novo método, denominado HRNet OCR W48 Frame Field para a extração de prédios por meio de imagens de sensoriamento remoto de altíssima resolução;

(e) Novo conjunto de dados rotulado para o treinamento de redes neurais convolucionais para a extração de prédios denominado *Brazilian Army Geographic Service Buildings Dataset*;

(f) Complemento para o QGIS para a construção e visualização de máscaras para o treinamento, denominado DeepLearningTools;

(g) *Framework* em Python, baseado no Tensorflow, denominado segmentation_models_trainer;

(h) *Framework* em Python, baseado no PyTorch, denominado pytorch_segmentation_models_trainer;

(i) Pesos dos treinamentos disponíveis nos repositórios de cada um dos *frameworks*; e

(j) Arquivos de configuração dos treinamentos supracitados, além das configurações das inferências;

Portanto, diante da pesquisa aqui apresentada, sugerem-se trabalhos futuros que testem os métodos propostos em outras áreas do Brasil. Além disso, pesquisas sobre misturas de conjuntos de dados e treinamentos com inicializações diferentes das que foram propostas são relevantes.

Ademais, são relevantes pesquisas utilizando arquiteturas diferentes combinadas tanto ao método Frame Field, quanto ao PolyMapper. Em particular, sugere-se o uso de *Vision Transformers*, como o SWIN Transformer V2 (LIU et al., 2021), em alternativa aos *backbones* dos referidos métodos.

Por fim, outra pesquisa interessante é o uso da nova técnica Deep Snakes (PENG et al., 2020) para extração de anotações em imagens, método este que os autores recentemente alegaram produzir resultados melhores que os atuais.

Em conclusão, espera-se que essa pesquisa possa ser aplicada na prática nas linhas de produção dos Centros de Geoinformação, braços produtivos da Diretoria de Serviço Geográfico, permitindo dessa forma que mais cartas 1:25.000 sejam produzidas anualmente para a sociedade brasileira.

8 Referências Bibliográficas

ABADI, M.; AGARWAL, A.; BARHAM, P.; BREVDO, E.; CHEN, Z.; CITRO, C.; CORRADO, G. S.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; GOODFELLOW, I.; HARP, A.; IRVING, G.; ISARD, M.; JIA, Y.; JOZEFOWICZ, R.; KAISER, L.; KUDLUR, M.; LEVENBERG, J.; MANé, D.; MONGA, R.; MOORE, S.; MURRAY, D.; OLAH, C.; SCHUSTER, M.; SHLENS, J.; STEINER, B.; SUTSKEVER, I.; TALWAR, K.; TUCKER, P.; VANHOUCKE, V.; VASUDEVAN, V.; VIéGAS, F.; VINYALS, O.; WARDEN, P.; WATTENBERG, M.; WICKE, M.; YU, Y.; ZHENG, X. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [S.l.], 2015. Software available from tensorflow.org. Disponível em: <https://www.tensorflow.org/>.

ACUNA, D.; LING, H.; KAR, A.; FIDLER, S. Efficient interactive annotation of segmentation datasets with polygon-rnn++. 2018.

ADARME, M. O.; FEITOSA, R. Q.; HAPP, P. N.; ALMEIDA, C. A. D.; GOMES, A. R. Evaluation of deep learning techniques for deforestation detection in the brazilian amazon and cerrado biomes from remote sensing imagery. *Remote Sensing*, v. 12, n. 6, 2020. ISSN 2072-4292. Disponível em: <https://www.mdpi.com/2072-4292/12/6/910>.

ALHASSAN, V.; HENRY, C.; RAMANNA, S.; STORIE, C. A deep learning framework for land-use/land-cover mapping and analysis using multispectral satellite imagery. *Neural Computing and Applications*, v. 32, n. 12, p. 8529–8544, 2020. ISSN 14333058. Disponível em: <https://doi.org/10.1007/s00521-019-04349-9>.

ANDERSON, T. W.; DARLING, D. A. Asymptotic Theory of Certain ”Goodness of Fit”Criteria Based on Stochastic Processes. *The Annals of Mathematical Statistics*, Institute of Mathematical Statistics, v. 23, n. 2, p. 193 – 212, 1952. Disponível em: <https://doi.org/10.1214/aoms/1177729437>.

ATHIWARATKUN, B.; FINZI, M.; IZMAILOV, P.; WILSON, A. G. There are many consistent explanations of unlabeled data: Why you should average. *arXiv preprint arXiv:1806.05594*, 2018.

AUDEBERT, N.; Le Saux, B.; LEFEVRE, S. Deep learning for classification of hyperspectral data: A comparative review. *IEEE Geoscience and Remote Sensing Magazine*, v. 7, n. 2, p. 159–173, 2019. ISSN 21686831.

AVBELJ, J.; MULLER, R.; BAMLER, R. A metric for polygon comparison and building extraction evaluation. *IEEE Geoscience and Remote Sensing Letters*, Institute of Electrical and Electronics Engineers Inc., v. 12, n. 1, p. 170–174, 2015. ISSN 1545598X.

BALL, J. E.; ANDERSON, D. T.; CHAN, C. S. Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, v. 11, n. 04, p. 1, 2017. ISSN 1931-3195.

BITTNER, K.; ADAM, F.; CUI, S.; KÖRNER, M.; REINARTZ, P. Building Footprint Extraction From VHR Remote Sensing Images Combined With Normalized DSMs Using Fused Fully Convolutional Networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, v. 11, n. 8, p. 2615–2629, 2018. ISSN 21511535.

BLASCHKE, T. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, Elsevier B.V., v. 65, n. 1, p. 2–16, 2010. ISSN 09242716. Disponível em: <http://dx.doi.org/10.1016/j.isprsjprs.2009.06.004>.

BOGUSZEWSKI, A.; BATORSKI, D.; ZIEMBA-JANKOWSKA, N.; ZAMBRZYCKA, A.; DZIEDZIC, T. LandCover.ai: Dataset for Automatic Mapping of Buildings, Woodlands and Water from Aerial Imagery. p. 1–14, 2020. Disponível em: <http://arxiv.org/abs/2005.02264>.

BORBA, P. *phborba/DeepLearningTools: First release*. Zenodo, oct 2020. Disponível em: <https://doi.org/10.5281/zenodo.4140680>.

BORBA, P. *phborba/segmentation_models_trainer: First Release*. Zenodo, sep 2020. Disponível em: <https://doi.org/10.5281/zenodo.4060390>.

BORBA, P. *phborba/pytorch_segmentation_models_trainer: Version 0.16.4*. Zenodo, dez. 2021. Disponível em: <https://doi.org/10.5281/zenodo.5761925>.

BORBA, P.; BIAS, E. D. S.; CORREIA, N.; ROIG, H. L. A review of remote sensing applications on very high-resolution imagery using deep learning-based semantic segmentation techniques. *International Journal of Advanced Engineering Research and Science*, v. 6495, p. 238–255, 2021.

BORBA, P.; DINIZ, F. de C.; SILVA, N. C. da; BIAS, E. de S. Building footprint extraction using deep learning semantic segmentation techniques: Experiments and results. In: *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*. [S.l.: s.n.], 2021. p. 4708–4711.

BRASIL. *Decreto-Lei nº 243, de 28 de fevereiro de 1967*. 1967. 2438 p. Disponível em: <https://www2.camara.leg.br/legin/fed/declei/1960-1969/decreto-lei-243-28-fevereiro-1967-376132-publicacaooriginal-1-pe.htmlhttps://www2.camara.leg.br/legin/fed/emecon/2016/emendaconstitucional-95-15-dezembro-2016-784029-publicacaooriginal-151558-pl.html{\%}0Ahttps://www2.camara.leg.br/legin/fed/decret/2006/decreto-5800-8-junho-2006-543167-publicacaooriginal-53181-pe.html{\%}0Ahttps:/>.

BRASIL. *Decreto nº 6.666, de 27 de novembro de 2008*. 2008. Disponível em: <http://www.planalto.gov.br/ccivil{_}03/{_}Ato2007-2010/2008/Decreto/D6666.htmhttp://www.inde.gov.br/images/inde/20@Decreto6666{_}27112008.pdf>.

BUSLAEV A. PARINOV, E. K. V. I. I. A.; KALININ, A. A. Albumentations: fast and flexible image augmentations. *ArXiv e-prints*, 2018.

CARRANZA-GARCÍA, M.; GARCÍA-GUTIÉRREZ, J.; RIQUELME, J. C. A framework for evaluating land use and land cover classification using convolutional neural networks. *Remote Sensing*, v. 11, n. 3, 2019. ISSN 20724292.

CASTREJÓN, L.; KUNDU, K.; URTASUN, R.; FIDLER, S. Annotating object instances with a polygon-rnn. In: *CVPR*. [S.l.: s.n.], 2017.

CHEN, G.; WENG, Q.; HAY, G. J.; HE, Y. Geographic object-based image analysis (GEOBIA): emerging trends and future opportunities. *GIScience and Remote Sensing*, Taylor & Francis, v. 55, n. 2, p. 159–182, 2018. ISSN 15481603. Disponível em: <https://doi.org/10.1080/15481603.2018.1426092>.

CHEN, J. X. The Evolution of Computing: AlphaGo. *Computing in Science and Engineering*, IEEE, v. 18, n. 4, p. 4–7, 2016. ISSN 15219615.

CHEN, L.-C.; ZHU, Y.; PAPANDREOU, G.; SCHROFF, F.; ADAM, H. 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. p. 801–818.

CHEN, Q.; WANG, L.; WU, Y.; WU, G.; GUO, Z.; WASLANDER, S. L. Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings. *ISPRS Journal of Photogrammetry and Remote Sensing*, Elsevier B.V., v. 147, p. 42–55, jan 2019. ISSN 09242716.

CHEN, Q.; WANG, L.; WU, Y.; WU, G.; GUO, Z.; WASLANDER, S. L. Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings. *ISPRS Journal of Photogrammetry and Remote Sensing*, Elsevier, v. 147, n. November 2018, p. 42–55, 2019. ISSN 09242716. Disponível em: <https://doi.org/10.1016/j.isprsjprs.2018.11.011>.

CHENG, D.; LIAO, R.; FIDLER, S.; URTASUN, R. *DARNet: Deep Active Ray Network for Building Segmentation*. 2019.

CHENG, G.; ZHOU, P.; HAN, J. Learning Rotation-Invariant Convolutional Neural Networks for Object Detection in VHR Optical Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, IEEE, v. 54, n. 12, p. 7405–7415, 2016. ISSN 01962892.

CLEVERT, D. A.; UNTERTHINER, T.; HOCHREITER, S. Fast and accurate deep network learning by exponential linear units (ELUs). *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, p. 1–14, 2016.

CONCAR. *Especificação Técnica para a Estruturação de Dados Geoespaciais Vetoriais (ET-EDGV)*. Brasília, DF: [s.n.], 2010. 246 p. Disponível em: <http://www.geoportal.eb.mil.br/images/PDF/ET{_}EDGV{_}Vs{_}2{_}1{_}3.pdf>.

DAMODARAN, B. B.; HÖHLE, J.; LEFÈVRE, S. Attribute profiles on derived features for urban land cover classification. *Photogrammetric Engineering and Remote Sensing*, v. 83, n. 3, p. 183–193, 2017. ISSN 00991112.

DATA, H.; CHEN, Y.; LIN, Z.; CHEN, Y.; LIN, Z.; ZHAO, X.; MEMBER, S. Deep Learning-Based Classification of Hyperspectral Data. IEEE, v. 7, n. June 2014, p. 1–14, 2015.

DIAKOGIANNIS, F. I.; WALDNER, F.; CACCETTA, P.; WU, C. ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data. 2019. Disponível em: <http://arxiv.org/abs/1904.00592>.

DOUGLAS, D. H.; PEUCKER, T. K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, University of Toronto Press, v. 10, n. 2, p. 112–122, 1973.

DSG. *Banco de Dados Geográficos do Exército*. Brasília, DF: [s.n.], 2010. Disponível em: <www.bdgex.eb.mil.br>.

DSG. *Especificação Técnica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG)*. Brasília: [s.n.], 2016. 1–94 p. Disponível em: <https://www.geoportal.eb.mil.br/portal/inde2>.

DSG. *Especificação Técnica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG)*. Brasília: [s.n.], 2016. 1–94 p. Disponível em: <https://www.geoportal.eb.mil.br/portal/inde2>.

DSG. *Especificação Técnica para Aquisição de Dados Geoespaciais Vetoriais (ET-ADGV) Versão 3.0, 1ª Edição*. Brasília, DF: [s.n.], 2018.

FACEBOOK. *Papers with Code*. 2018. Disponível em: <https://www.paperswithcode.com>.

FALCON, W. Pytorch lightning. *GitHub. Note: https://-github.com/PyTorchLightning/pytorch-lightning*, v. 3, 2019.

GIRARD, N.; SMIRNOV, D.; SOLOMON, J.; TARABALKA, Y. REGULARIZED BUILDING SEGMENTATION BY FRAME FIELD LEARNING. *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, p. 1805–1808, 2020.

GIRARD, N.; SMIRNOV, D.; SOLOMON, J.; TARABALKA, Y. Polygonal building extraction by frame field learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2021. p. 5891–5900.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016.

GUIRADO, E.; TABIK, S.; ALCARAZ-SEGURA, D.; CABELLO, J.; HERRERA, F. Deep-learning Versus OBIA for scattered shrub detection with Google Earth Imagery: Ziziphus lotus as case study. *Remote Sensing*, v. 9, n. 12, p. 1–22, 2017. ISSN 20724292.

HE, K.; GKIOXARI, G.; DOLLÁR, P.; GIRSHICK, R. Mask r-cnn. In: *Proceedings of the IEEE international conference on computer vision*. [S.l.: s.n.], 2017. p. 2961–2969.

HOBEL, H.; ABDALLA, A.; FOGLIARONI, P.; FRANK, A. U. A semantic region growing algorithm: Extraction of urban settings. In: *Lecture Notes in Geoinformation and Cartography*. [S.l.: s.n.], 2015. ISBN 9783319167862. ISSN 18632351.

HOESER, T.; BACHOFER, F.; KUENZER, C. Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review—Part II: Applications. *Remote Sensing*, v. 12, n. 18, p. 3053, 2020. ISSN 2072-4292.

HOESER, T.; KUENZER, C. Object detection and image segmentation with deep learning on Earth observation data: A review-part I: Evolution and recent trends. *Remote Sensing*, v. 12, n. 10, 2020. ISSN 20724292.

HU, F.; XIA, G. S.; HU, J.; ZHANG, L. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, v. 7, n. 11, p. 14680–14707, 2015. ISSN 20724292.

HU, J.; SHEN, L.; ALBANIE, S.; SUN, G.; WU, E. Squeeze-and-Excitation Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 42, n. 8, p. 2011–2023, 2020. ISSN 19393539.

HUNDT, A.; JAIN, V.; HAGER, G. D. *sharpDARTS: Faster and More Accurate Differentiable Architecture Search*. 2019.

IOFFE, S.; SZEGEDY, C. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015.

ISPRS. *2D Semantic Labeling Contest*. 2012. Disponível em: <http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html>.

IZMAILOV, P.; PODOPRIKHIN, D.; GARIPOV, T.; VETROV, D.; WILSON, A. G. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

JOZDANI, S. E.; JOHNSON, B. A.; CHEN, D. Comparing Deep Neural Networks , Ensemble Classifiers , and Support Vector Machine Algorithms. *Remote Sensing*, v. 11, n. 1, p. 1–24, 2019.

KEMKER, R.; SALVAGGIO, C.; KANAN, C. Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 145, p. 60–77, 2018. ISSN 09242716.

KENDALL, M. G. A new measure of rank correlation. *Biometrika*, JSTOR, v. 30, n. 1/2, p. 81–93, 1938.

KHANAL, N.; UDDIN, K.; MATIN, M. A.; TENNESON, K. Automatic detection of spatiotemporal urban expansion patterns by fusing OSM and Landsat data in Kathmandu. *Remote Sensing*, v. 11, n. 19, 2019. ISSN 20724292.

KOLMOGOROV, A. Sulla determinazione empirica di una lgge di distribuzione. *Inst. Ital. Attuari, Giorn.*, v. 4, p. 83–91, 1933.

KUCHARCZYK, M.; HAY, G. J.; GHAFFARIAN, S.; HUGENHOLTZ, C. H. Geographic Object-Based Image Analysis: A Primer and Future Directions. *Remote Sensing*, v. 12, n. 12, p. 2012, 2020. ISSN 2072-4292.

KULKARNI, T.; VENUGOPAL, N. Automatic semantic segmentation for change detection in remote sensing images. *Advances in Intelligent Systems and Computing*, Springer US, v. 705, n. 0123456789, p. 337–344, 2020. ISSN 21945357. Disponível em: <https://doi.org/10.1007/s11063-019-10174-x>.

LANARAS, C.; BIOUCAS-DIAS, J.; GALLIANI, S.; BALTSAVIAS, E.; SCHINDLER, K.; SENSING, R.; ZURICH, E. Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network. 2018. Disponível em: <https://doi.org/10.1016/j.isprsjprs.2018.09.018>.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, p. 436–444, 2015. ISSN 14764687.

LI, J.; LI, X.; XIE, T. Morphing of building footprints using a turning angle function. *ISPRS International Journal of Geo-Information*, MDPI AG, v. 6, n. 6, jun 2017. ISSN 22209964.

LI, K.; HU, X.; JIANG, H.; SHU, Z.; ZHANG, M. Attention-Guided Multi-Scale Segmentation Neural Network for Interactive Extraction of Region Objects from High-Resolution Satellite Imagery. 2020.

LI, M.; ZANG, S.; ZHANG, B.; LI, S.; WU, C. A review of remote sensing image classification techniques: The role of Spatio-contextual information. *European Journal of Remote Sensing*, Associazione Italiana di Telerilevamento, v. 47, n. 1, p. 389–411, jun 2014. ISSN 22797254.

LI, Z.; ARORA, S. An exponential learning rate schedule for deep learning. *arXiv preprint arXiv:1910.07454*, 2019.

LI, Z.; WEGNER, J. D.; LUCCHI, A. *Topological Map Extraction from Overhead Images*. 2019.

LIN, T. Y.; DOLLÁR, P.; GIRSHICK, R.; HE, K.; HARIHARAN, B.; BELONGIE, S. Feature pyramid networks for object detection. In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*. [S.l.: s.n.], 2017. v. 2017-Janua, p. 936–944. ISBN 9781538604571.

LING, H.; GAO, J.; KAR, A.; CHEN, W.; FIDLER, S. Fast interactive object annotation with curve-gcn. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2019. p. 5257–5266.

LIU, H.; LIU, F.; FAN, X.; HUANG, D. *Polarized Self-Attention: Towards High-quality Pixel-wise Regression*. 2021.

LIU, Q.; KAMPFFMEYER, M.; JESSEN, R.; SALBERG, A.-B. Dense Dilated Convolutions Merging Network for Land Cover Classification. n. March, 2020. Disponível em: <http://arxiv.org/abs/2003.04027{\%}0Ahttp://dx.doi.org/10.1109/TGRS.2020.2976658>.

LIU, Y.; PANG, C.; ZHAN, Z.; ZHANG, X.; YANG, X. Building Change Detection for Remote Sensing Images Using a Dual Task Constrained Deep Siamese Convolutional Network Model. 2019. Disponível em: <http://arxiv.org/abs/1909.07726>.

LIU, Z.; LIN, Y.; CAO, Y.; HU, H.; WEI, Y.; ZHANG, Z.; LIN, S.; GUO, B. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

LOKHAT, I.; TOUYA, G. Enhancing building footprints with squaring operations. *Journal of Spatial Information Science*, v. 13, n. 2016, p. 33–60, 2016. ISSN 1948660X.

LONG, Y.; XIA, G. S.; LI, S.; YANG, W.; YANG, M. Y.; ZHU, X. X.; ZHANG, L.; LI, D. DiRS: On creating benchmark datasets for remote sensing image interpretation. *arXiv*, p. 1–22, 2020.

LORENSEN, W. E.; CLINE, H. E. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, ACM New York, NY, USA, v. 21, n. 4, p. 163–169, 1987.

LOSHCHILOV, I.; HUTTER, F. *SGDR: Stochastic Gradient Descent with Warm Restarts*. 2017.

MA, L.; LI, M.; MA, X.; CHENG, L.; DU, P.; LIU, Y. *A review of supervised object-based land-cover image classification*. 2017. 277–293 p.

MA, L.; LIU, Y.; ZHANG, X.; YE, Y.; YIN, G.; JOHNSON, B. A. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, Elsevier, v. 152, n. March, p. 166–177, 2019. ISSN 09242716. Disponível em: <https://doi.org/10.1016/j.isprsjprs.2019.04.015>.

MAGGIORI, E.; TARABALKA, Y.; CHARPIAT, G.; ALLIEZ, P. Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark. *International Geoscience and Remote Sensing Symposium (IGARSS)*, v. 2017-July, p. 3226–3229, 2017.

MBOGA, N.; GEORGANOS, S.; GRIPPA, T.; LENNERT, M.; VANHUYSSE, S.; WOLFF, E. Fully convolutional networks and geographic object-based image analysis for the classification of VHR imagery. *Remote Sensing*, v. 11, n. 5, 2019. ISSN 20724292.

MNIH, V. Machine Learning for Aerial Image Labeling. *PhD Thesis*, p. 109, 2013.

MOHANTY, S. P.; CZAKON, J.; KACZMAREK, K. A.; PYSKIR, A.; TARASIEWICZ, P.; KUNWAR, S.; ROHRBACH, J.; LUO, D.; PRASAD, M.; FLEER, S. et al. Deep learning for understanding satellite imagery: An experimental survey. *Frontiers in Artificial Intelligence*, Frontiers Media SA, v. 3, 2020.

MUSYAROFAH; SCHMIDT, V.; KADA, M. Object detection of aerial image using mask-region convolutional neural network (mask R-CNN). In: *IOP Conference Series: Earth and Environmental Science*. [S.l.]: Institute of Physics Publishing, 2020. v. 500, n. 1. ISSN 17551315.

NAJAFABADI, M. M.; VILLANUSTRE, F.; KHOSHGOFTAAR, T. M.; SELIYA, N.; WALD, R.; MUHAREMAGIC, E. Deep learning applications and challenges in big data analytics. *Journal of Big Data*, v. 2, p. 1, 2015.

NESTEROV, Y. *Introduction to convex optimization: A basic course*. [S.l.]: Springer, 2004.

NEUPANE, B.; HORANONT, T.; ARYAL, J. Deep learning-based semantic segmentation of urban features in satellite images: A review and meta-analysis. *Remote Sensing*, v. 13, p. 1–41, 2021. ISSN 20724292.

PAN, B.; XU, X.; SHI, Z.; ZHANG, N.; LUO, H.; LAN, X. DSSNet: A Simple Dilated Semantic Segmentation Network for Hyperspectral Imagery Classification. *IEEE Geoscience and Remote Sensing Letters*, IEEE, p. 1–5, 2020. ISSN 1545-598X.

PARTOVI, T.; BAHMANYAR, R.; KRAUS, T.; REINARTZ, P. Building Outline Extraction Using a Heuristic Approach Based on Generalization of Line Segments. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, v. 10, n. 3, p. 933–947, 2017. ISSN 21511535.

PASZKE, A.; GROSS, S.; MASSA, F.; LERER, A.; BRADBURY, J.; CHANAN, G.; KILLEEN, T.; LIN, Z.; GIMELSHEIN, N.; ANTIGA, L.; DESMAISON, A.; KOPF, A.; YANG, E.; DEVITO, Z.; RAISON, M.; TEJANI, A.; CHILAMKURTHY, S.; STEINER, B.; FANG, L.; BAI, J.; CHINTALA, S. Pytorch: An imperative style, high-performance deep learning library. In: WALLACH, H.; LAROCHELLE, H.; BEYGELZIMER, A.; ALCHé-BUC, F. d'; FOX, E.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019. p. 8024–8035. Disponível em: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.

PENG, S.; JIANG, W.; PI, H.; LI, X.; BAO, H.; ZHOU, X. Deep snake for real-time instance segmentation. In: *CVPR*. [S.l.: s.n.], 2020.

QGIS Development Team. *QGIS Geographic Information System*. [S.l.], 2009. Disponível em: <http://qgis.org>.

QIN, M.; MAVROMATIS, S.; HU, L.; ZHANG, F.; LIU, R.; SEQUEIRA, J.; DU, Z. Remote Sensing Single-Image Resolution Improvement Using A Deep Gradient-Aware Network with Image-Specific Enhancement. *Remote Sensing*, MDPI AG, v. 12, n. 5, p. 758, feb 2020. ISSN 2072-4292. Disponível em: <https://www.mdpi.com/2072-4292/12/5/758>.

RAMER, U. An iterative procedure for the polygonal approximation of plane curves. *Computer graphics and image processing*, Elsevier, v. 1, n. 3, p. 244–256, 1972.

RIBA D. MISHKIN, J. S. D. P. F. M.-N. E.; BRADSKI, G. A survey on kornia: an open source differentiable computer vision library for pytorch. In: . [S.l.: s.n.], 2020.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. [S.l.: s.n.], 2015. v. 9351, p. 234–241. ISBN 9783319245737. ISSN 16113349.

SALVETTI, F.; MAZZIA, V.; KHALIQ, A.; CHIABERGE, M. Multi-Image Super Resolution of Remotely Sensed Images Using Residual Attention Deep Neural Networks. *Remote Sensing*, v. 12, n. 14, p. 2207, jul 2020. ISSN 2072-4292. Disponível em: <https://www.mdpi.com/2072-4292/12/14/2207>.

SHAPIRO, L. *Computer vision*. Upper Saddle River, NJ: Prentice Hall, 2001. ISBN 0130307963.

SHORTEN, C.; KHOSHGOFTAAR, T. M. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, Springer International Publishing, v. 6, n. 1, 2019. ISSN 21961115. Disponível em: <https://doi.org/10.1186/s40537-019-0197-0>.

SIMONYAN, K.; ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015.

SMITH, L. N. *A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay*. 2018.

SONG, A.; CHOI, J. Fully Convolutional Networks with Multiscale 3D Filters and Transfer Learning for Change Detection in High Spatial Resolution Satellite Images. *Remote Sensing*, Multidisciplinary Digital Publishing Institute, v. 12, n. 5, p. 799, mar 2020. ISSN 2072-4292. Disponível em: <https://www.mdpi.com/2072-4292/12/5/799>.

SPACENET. *SpaceNet on Amazon Web Services (AWS)*. 2018. Disponível em: <https://spacenet.ai/datasets/>.

SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, v. 15, n. 56, p. 1929–1958, 2014. Disponível em: <http://jmlr.org/papers/v15/srivastava14a.html>.

SU, W.; LI, J.; CHEN, Y.; LIU, Z.; ZHANG, J.; LOW, T. M.; SUPPIAH, I.; HASHIM, S. A. M. Textural and local spatial statistics for the object□oriented classification of urban areas using high resolution imagery. *International Journal of Remote Sensing*, Taylor Francis, v. 29, n. 11, p. 3105–3117, 2008. Disponível em: <https://doi.org/10.1080/01431160701469016>.

SZEGEDY, C.; VANHOUCKE, V.; IOFFE, S.; SHLENS, J.; WOJNA, Z. Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 2818–2826.

TAN, M.; LE, Q. V. *EfficientNet: Rethinking model scaling for convolutional neural networks*. [S.l.], 2019. v. 2019-June, 10691–10700 p.

TAO, A.; SAPRA, K.; CATANZARO, B. Hierarchical Multi-Scale Attention for Semantic Segmentation. 2020. Disponível em: <http://arxiv.org/abs/2005.10821>.

Thanh Noi, P.; KAPPAS, M. Comparison of Random Forest, k-Nearest Neighbor, and Support Vector Machine Classifiers for Land Cover Classification Using Sentinel-2 Imagery. *Sensors (Basel, Switzerland)*, v. 18, n. 1, 2017. ISSN 14248220.

Van Etten, A.; LINDENBAUM, D.; BACASTOW, T. SpaceNet: A remote sensing dataset and challenge series. *arXiv*, 2018.

WANG, Q.; ZHANG, X.; CHEN, G.; DAI, F.; GONG, Y.; ZHU, K. Change detection based on Faster R-CNN for high-resolution remote sensing images. *Remote Sensing Letters*, Taylor and Francis Ltd., v. 9, n. 10, p. 923–932, oct 2018. ISSN 21507058.

WISEMAN, G.; KORT, J.; WALKER, D. Quantification of shelterbelt characteristics using high-resolution imagery. *Agriculture, ecosystems & environment*, Elsevier, v. 131, n. 1-2, p. 111–117, 2009.

YADAN, O. *Hydra - A framework for elegantly configuring complex applications*. 2019. Github. Disponível em: <https://github.com/facebookresearch/hydra>.

YAKUBOVSKIY, P. *Segmentation Models*. [S.l.]: GitHub, 2019. <https://github.com/qubvel/segmentation_models>.

YAKUBOVSKIY, P. *Segmentation Models Pytorch*. [S.l.]: GitHub, 2020. <https://github.com/qubvel/segmentation_models.pytorch>.

YANG, H.; WU, P.; YAO, X.; WU, Y.; WANG, B.; XU, Y. Building extraction in very high resolution imagery by dense-attention networks. *Remote Sensing*, v. 10, n. 11, p. 1–16, 2018. ISSN 20724292.

ZHANG, C.; WEI, S.; JI, S.; LU, M. Detecting large-scale urban land cover changes from very high resolution remote sensing images using CNN-based classification. *ISPRS International Journal of Geo-Information*, v. 8, n. 4, 2019. ISSN 22209964.

ZHANG, F.; NAUATA, N.; FURUKAWA, Y. *Conv-MPN: Convolutional Message Passing Neural Network for Structured Outdoor Architecture Reconstruction*. 2021.

ZHANG, H.; CISSE, M.; DAUPHIN, Y. N.; LOPEZ-PAZ, D. *mixup: Beyond Empirical Risk Minimization*. 2018.

ZHANG, H.; WU, C.; ZHANG, Z.; ZHU, Y.; LIN, H.; ZHANG, Z.; SUN, Y.; HE, T.; MUELLER, J.; MANMATHA, R.; LI, M.; SMOLA, A. ResNeSt: Split-Attention Networks. *arXiv*, 2020.

ZHANG, L.; ZHANG, L.; DU, B. Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, v. 4, n. 2, p. 22–40, 2016. ISSN 21686831.

ZHAO, H.; SHI, J.; QI, X.; WANG, X.; JIA, J. Pyramid scene parsing network. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, v. 2017-January, p. 6230–6239, 2017.

ZHAO, W.; PERSELLO, C.; STEIN, A. Building outline delineation: From aerial images to polygons with an improved end-to-end learning framework. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 175, p. 119–131, 2021. ISSN 0924-2716. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0924271621000551>.

ZHU, X. X.; TUIA, D.; MOU, L.; XIA, G. S.; ZHANG, L.; XU, F.; FRAUNDORFER, F. *Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources*. [S.l.]: Institute of Electrical and Electronics Engineers Inc., dec 2017. 8–36 p.

A Anexos do artigo do capítulo 6

Appendices

Appendix A.1 Tables with references of each detailed analysis

Table 15: Surveyed papers grouped by dataset.

Dataset	Papers	Number of Papers
WHU Aerial Building Dataset	[8, 9, 11, 48, 56, 89, 58, 93, 50, 95, 46, 81, 90, 73, 85, 60, 74, 51, 99, 75, 62, 91, 92]	23
Custom	[10, 76, 100, 77, 53, 68, 61, 96, 71, 65, 63, 87, 49, 102, 73, 103, 66, 101, 55, 82]	20
INRIA Aerial Image Labeling Dataset	[8, 11, 13, 94, 52, 56, 53, 84, 58, 46, 79, 54, 90, 47, 85, 83]	16
Massachusetts Buildings Dataset	[13, 14, 94, 52, 15, 80, 57, 44, 84, 59, 81, 90, 67, 75, 62, 91]	16
ISPRS Potsdam	[88, 64, 12, 105, 57, 89, 45, 59, 86, 72, 60, 98]	12
ISPRS Vaihingen	[88, 64, 70, 80, 45, 68, 72, 73, 85, 60]	10
DREAM-B	[104, 47, 67]	3
SpaceNet Building Dataset	[69, 50]	2
Open Cities Dataset	[97]	1
AISD Dataset	[98]	1
Urban 3-D Challenge Dataset	[50]	1
DataPlus Dataset	[64]	1
CrowdAI	[97]	1
DB UAV Rural Building Dataset	[95]	1
AIRS Dataset	[78]	1

Table 16: Surveyed papers grouped by chosen frameworks.

Framework	Papers	Number of papers
PyTorch	[18, 111, 98, 99, 53, 59, 80, 95, 78, 89, 17, 51, 103, 106, 57, 48, 58, 87, 83, 56, 72, 109, 49, 127, 45, 66, 122, 82]	28
no info	[67, 94, 55, 61, 119, 68, 19, 70, 124, 74, 20, 11, 118, 22, 105, 113, 104, 71, 16, 93, 121, 101]	22
Tensorflow	[81, 65, 92, 100, 79, 90, 125, 46, 12, 52, 86, 13, 110, 97]	14
Keras with Tensorflow	[108, 44, 62, 75, 91, 112, 50, 84, 8, 10, 88, 9]	12
Caffe	[114, 60, 115, 77, 120, 14, 76, 64]	8
Keras	[107, 47, 63, 85, 15, 54, 69, 102]	8
Matlab	[117, 123, 126, 116]	4
ENVI with Tensorflow	[96]	1
LISA + SURFSara	[21]	1
MXNet	[73]	1

Table 18: Architecture groups that were grouped more than once to some particular backbone.

Architecture Group	Backbone	Papers
U-Net based	ResNet-34	[99, 58, 82]
U-Net based	VGG-16	[77]
U-Net based	Custom	[74, 94, 84, 85, 83, 13, 55, 65, 71]
U-Net based	Custom (ESPC module)	[67]
U-Net based	Custom, Siamese	[8]
U-Net based	NASNet	[47]

Continued on next page

Table 18: Architecture groups that were grouped more than once to some particular backbone.

Architecture Group	Backbone	Papers
U-Net based	NASNet-Mobile	[104]
U-Net based	ResNet	[88, 59, 10]
U-Net based	VGG	[44]
U-Net based	no info	[96]
U-Net based	classical from original papers	[69]
U-Net based	Xception	[80]
CNN based	InceptionV3, MobileNet	[68]
CNN based	Custom	[73, 105]
Custom	VGG-19 and SegNet	[72]
Custom	VGG-16	[60]
CNN based	ResNet-101	[57]
CNN based	RetinaNet	[100]
CNN based	classical from original papers	[61]
Custom	Custom	[70, 91]
Custom	Modified DarkNet-53	[46]
Custom	ResNet-50	[95]
FPN based	ResNet-50	[62]
FPN based	ResNet-101	[89, 97]
FPN based	ResNeXt-50	[54]
FPN based	Custom	[93]
FCN based	VGG-19	[79]
FCN based	Custom VGG-16	[9]
FCN based	Custom	[52, 76]
SegNet	Custom	[63, 49]
SegNet	VGG-16	[86]
MAP-Net	Custom	[50]

Continued on next page

Table 18: Architecture groups that were grouped more than once to some particular backbone.

Architecture Group	Backbone	Papers
DeepLabV3+	ResNet-101	[103]
MAP-Net	Custom, MAP-Net based	[90]
DeepLabV3+	Xception	[45]
DenseNet	Custom	[15, 12]
PSPNet	ResNet-50	[78]
FC-DenseNet	Custom	[87]
Web-Net	ResNeXt-50	[56]
U2-Net	Residual U-Blocks	[51]
D-LinkNet	Custom	[66]
DE-Net	Custom ResNet	[48]
GRRNet	ResNet-50	[64]
RCF Network	Custom	[14]
HRNetV2	Custom	[92]
LSTM	Custom	[101]
MHA-Net	Custom	[75]
MTPA-Net	Custom	[53]
NFSNet	ResNet-18	[98]
O-GAN	Custom	[102]
SRI	ResNet-101 with dilated convolutions	[11]
BMFR-Net (U-Net + FPN)	Custom	[81]

Table 17: Papers with code. All these papers relate to Building footprint extraction, except [107], wich is a change detection research. [52, 65] employ GAN methods, while all the other use semantic segmentation using CNNs. All papers have raster output, except [103], that outputs vector polygons.

Reference	Code Link
[64]	`https://github.com/CHUANQIFENG/GRRNet`
[76]	`https://gitlab.com/ksenia_bittner/fused-fcn4s`
[12]	`https://github.com/shenhuqiji/DAN`
[52]	`https://github.com/lixiang-ucas/Building-A-Nets`
[50]	`https://github.com/lehaifeng/MAPNet`
[65]	`https://github.com/affinelayer/pix2pix-tensorflow`
[107]	`https://github.com/Raveerat-titech/CORN`
[81]	`https://github.com/RanKoala/BMFR-Net`
[90]	`https://github.com/liaochengcsu/jlcs-building-extracion`
[47]	`https://github.com/yangnaisen/GeohashNet`
[103]	`https://github.com/LBNL-ETA/AutoBFE`
[60]	`https://github.com/CHUANQIFENG/ALRNet`
[99]	`https://github.com/HaonanGuo/SG-EPUNet`

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[88]	RGB, nDSM, NDVI	Raster	ISPRS Vaihingen, ISPRS Potsdam	U-Net	ResNet	Cross Entropy loss	no info	Precision, Recall, F1 Score
[8]	RGB	Raster	WHU Aerial Building dataset, INRIA Aerial Image Labeling Dataset	U-Net	Custom, Siamese	no info	random color	IoU, Precision, Recall
[64]	RGB, LiDAR, NIR	Raster	DataPlus dataset, ISPRS Vaihingen, ISPRS Potsdam	GRRNet	ResNet-50	Weighted Cross Entropy Loss	random rotation with mirrowing	Overall Accuracy, mIoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[9]	RGB	Raster	WHU Aerial Building dataset	FCN	Custom VGG-16	Joint loss (combination of Cross Entropy loss evaluated at each scale output)	Relative radiometric correction using Wallis filtering	IoU, Precision and Recall
[10]	RGB	Raster	Custom	U-Net	ResNet	Cross Entropy Loss	not used	Precision, Recall, F1 Score, Kappa, Overall Accuracy
[11]	RGB	Raster	INRIA Aerial Image Labeling Dataset, WHU Aerial Building Dataset	SRI	ResNet-101 with dilated convolutions	Lovasz loss	random scale, random color, random rotation, random flip	Precision, Recall, Average Precision, IoU
[69]	RGB	Vector	SpaceNet Building Dataset	U-Net	classical from original papers	Binary Cross Entropy Loss	random rotation, random scale	IoU, Precision, Recall, F1 Score

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[76]	RGB, nDSM, PAN	Raster	Custom	FCN	Custom	Softmax Cross Entropy Loss	no info	Mean Accuracy, Overall accuracy, mIoU, IoU, F1 Score
[12]	RGB	Raster	ISPRS Potsdam	DenseNet	Custom	Softmax Cross Entropy Loss	random flip	IoU, Precision, Recall, F1 Score, Quality
[13]	RGB	Raster	Massachusetts Buildings dataset, INRIA Aerial Image Labeling Dataset	BRRNet (Custom U-Net)	Custom	Binary Cross Entropy, Dice Loss	NaN	IoU, Precision, Recall, F1 Score
[14]	RGB	Raster	Massachusetts Buildings dataset	RCF Network	Custom	no info	no info	Precision, Recall, F1 Score

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[94]	RGB	Raster	Massachusetts Buildings dataset, INRIA Aerial Image Labeling Dataset	U-Net	Custom	Cross Entropy Loss, Multi-Scale L1 loss	random flip, random rotation	Accuracy, IoU, Precision, Recall, F1 Score
[52]	RGB	Raster	Massachusetts Buildings dataset, INRIA Aerial Image Labeling Dataset	FCN	Custom	Custom Loss	random flip	Precision, Recall, Breakeven Score, Relaxed F1 Score
[15]	RGB	Raster	Massachusetts Buildings dataset	DenseNet	Custom	no info	random rotation	F1 Score, Precision, Recall
[?]	LiDAR, RGB, DTM	Raster	ISPRS Vaihingen	Custom	Custom	Negative Log-Likelihood Loss	no info	Accuracy

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[105]	RGB	Raster	ISPRS Potsdam	SVM + CNNs	Custom	no info	no info	Recall, Precision, F1 Score
[48]	RGB	Raster	WHU Aerial Building dataset	DE-Net	Custom ResNet	Dice Binary Cross Entropy loss	random flip, random rotation	Precision, Recall, F1 Score, IoU
[100]	RGB, LiDAR	Raster	Custom	Mask R-CNN	RetinaNet	Joint loss (focal loss, smooth L1 loss)	no info	Precision, Recall, Mean Average Precision, Accuracy, F1 Score
[77]	RGB	Raster	Custom	Adapted U-Net	VGG-16	Logistic loss	no info	Mean Accuracy, Overall accuracy, mIoU, Precision, Recall, IoU, F1 Score

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[56]	RGB	Raster	INRIA Aerial Image Labeling Dataset, WHU Aerial Building Dataset	Web-Net	ResNeXt-50	Joint loss (weighted binary cross entropy loss, dice loss)	no info	IoU, Accuracy
[53]	RGB	Raster	INRIA Aerial Image Labeling Dataset, Custom	MTPA-Net	Custom	Binary Cross Entropy loss, Lovasz-Softmax loss	random rotation, random scale, random crop, random flip	IoU, Accuracy, Precision, Recall, F1 Score
[80]	RGB	Raster	Massachusetts Buildings dataset, ISPRS Vaihingen	U-Net	Xception	Joint loss (weighted probability log sum)	random flip, random rotation	Accuracy, F1 Score
[57]	RGB	Raster	ISPRS Potsdam, Massachusets Buildings dataset	L-GCNN	ResNet-101	Joint loss (normalized cross entropy loss, L2 norm loss, PGM loss)	random scaler, random rotation, random flip	Accuracy, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[44]	RGB	Raster	Massachusetts Buildings dataset	SegNet + U-Net	VGG	Binary Cross Entropy	no info	IoU, Precision, Recall, F1 Score, Overall Accuracy
[89]	RGB	Raster	ISPRS Potsdam, WHU Aerial Building dataset	Custom FPN	ResNet-101	EALoss	not used	Precision, Recall, F1 Score, IoU
[84]	RGB	Vector	INRIA Aerial Image Labeling Dataset, Massachusets Buildings dataset	MFCNN (U-Net based, with Pyramid Aggregation Unit)	Custom	Joint loss (cross entropy loss, dice loss)	random crop, random rotation, random noise	Overall Accuracy, Precision, Recall, F1 Score, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[58]	RGB	Raster	WHU Aerial Building dataset, INRIA Aerial Image Labeling Dataset	U-Net	ResNet-34	Boundary-Aware Perceptual Loss	no info	Accuracy, IoU, Precision, Recall, F1 Score
[104]	RGB	Raster	DREAM-B	U-Net	NASNet-Mobile	no info	random color, random rotation, random flip	Precision, Recall, F1 Score, Overall Accuracy, Kappa
[93]	RGB	Raster	WHU Aerial Building dataset	FPN	Custom	Binary Cross Entropy Loss, Joint Loss (alpha balance focal loss, centerness loss, mask loss)	random rotation, random flip	Precision, Recall, Average Precision

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[45]	RGB	Raster	ISPRS Potsdam, ISPRS Vaihingen	DeepLabV3+	Xception	Cross Entropy Loss	random rotation	Overall Accuracy, Detection Accuracy, False Alarm Rate, mIoU
[68]	RGB	Raster	ISPRS Vaihingen, Custom	CNN	InceptionV3, MobileNet	no info	no info	Precision, Recall, F1 Score, Overall Accuracy, Kappa
[61]	RGB	Raster	Custom	Mask R-CNN	classical from original papers	Cross Entropy Loss	random color, random rotation, random flip	Precision, Recall, Average Precision, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[50]	RGB	Raster	WHU Aerial Building dataset, SpaceNet Building Dataset, Urban 3-D Challenge dataset	MAP-Net	Custom	Sigmoid Loss	random rotation, random flip	Precision, Recall, F1 Score, IoU
[78]	RGB	Vector	AIRS Dataset	PSPNet	ResNet-50	Cross Entropy Loss, Bi-projection loss, Relative shape loss	no info	IoU, VertexF, VertexP, VertexR

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[95]	RGB	Raster	WHU Aerial Building dataset, DB UAV Rural Building Dataset	Custom	ResNet-50	Binary Cross Entropy	random scale, random rotation, random flip	IoU, Precision, Recall, F1 Score
[96]	RGB, LiDAR	Raster, Vector	Custom	U-Net	no info	Weighted Binary Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score, IoU
[59]	RGB	Raster	Massachusetts Buildings dataset, ISPRS Potsdam	U-Net	ResNet	no info	no info	Overall Accuracy, Precision, Recall, F1 Score
[86]	RGB, NIR, DSM	Raster, Vector	ISPRS Potsdam	SegNet	VGG-16	no info	random scale, random rotation, random flip	Overall Accuracy, Precision, Recall, F1 Score, IoU, mIoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[71]	RGB	Raster	Custom	UNet-AP	Custom	Categorical Cross Entropy loss	no info	Accuracy, IoU, Precision, Recall, F1 Score
[46]	RGB	Raster	INRIA Aerial Image Labeling Dataset, WHU Aerial Building Dataset	Custom	Modified DarkNet-53	Joint loss (cross entropy loss and, boundary loss)	random flip	IoU, Precision, Recall, F1 Score
[65]	RGB	Raster	Custom	U-Net on generator, PatchGAN discriminator	Custom	GAN Loss, L1 Loss	no info	Precision (Completeness), Recall (Correctness), F1 Score
[79]	RGB	Raster	INRIA Aerial Image Labeling Dataset	Custom FCN	VGG-19	Cross Entropy Loss	no info	IoU, Precision, Recall, F1 Score

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[97]	RGB	Vector	CrowdAI, Open Cities dataset	FPN	ResNet-101	Cross Entropy Loss, Smooth L1 Loss	no info	PoLiS, IoU, Average Precision
[54]	RGB	Raster	INRIA Aerial Image Labeling Dataset	FPN	ResNeXt-50	Binary Cross Entropy, BCE Dice Loss	random rotation, random flip	IoU, Accuracy, Combined
[63]	RGB	Raster	Custom	SegNet	Custom	Binary Cross Entropy	random crop, random flip, random rotate, random color	IoU, Precision, Recall, F1 Score, Overall Accuracy
[72]	RGB + DSM	Raster	ISPRS Vaihingen, ISPRS Potsdam	Custom	VGG-19 and SegNet	Cross Entropy Loss	random flip	IoU, Precision, Recall, F1 Score
[81]	RGB	Raster	WHU Aerial Building dataset, Massachusets Buildings dataset	BMFR-Net (U-Net + FPN)	Custom	Dice Loss, Binary Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[87]	RGB, nDSM, tDSM, True-DOP, DFK	Raster	Custom	FC-DenseNet	Custom	Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score, IoU
[90]	RGB	Raster	INRIA Aerial Image Labeling Dataset, WHU Aerial Building dataset, Massachusets Buildings dataset	MAP-Net	Custom, MAP-Net based	Joint loss (cross entropy loss, dice loss)	random flip, random rotation	Precision, Recall, F1 Score, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[49]	RGB	Raster	Custom	SegNet	Custom	L1 Loss, Mean Square Error Loss, Focal Loss, Binary Cross Entropy Loss, Focal Loss	no info	F1 Score, IoU, Kappa
[102]	RGB	Raster	Custom	O-GAN	Custom	Orthogonality Loss	no info	IoU
[47]	RGB	Raster	INRIA Aerial Image Labeling Dataset, DREAM-B	GeoHashNet = U-Net + GeoHash	NASNet	no info	random flip, random rotation, random color	mIoU, Overall Accuracy, Precision, Recall, F1 Score

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[73]	RGB	Raster	ISPRS Vaihingen, WHU Aerial Building dataset, Custom	EMU-CNN	Custom	Joint loss (negative log softmax as classification loss, smooth L1 as Bbox regression loss, negative log softmax as segmentation loss)	no info	Mean Average Precision, Precision, Recall
[103]	RGB	Vector	Custom	DeepLabV3+	ResNet-101	Joint loss (cross entropy loss, dice loss)	random flip, random color	Precision, Recall, F1 Score, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[85]	RGB	Raster	ISPRS Vaihingen, WHU Aerial Building dataset, INRIA Aerial Image Labeling Dataset	U-Net	Custom	Joint loss (combination of different losses, including Weighted Cross Entropy, Binary Cross Entropy and Smooth L1 Loss)	random scale, random rotation, random flip	IoU, Precision, Recall, F1 Score
[60]	RGB	Raster	ISPRS Potsdam, ISPRS Vaihingen, WHU Aerial Building dataset	Custom	VGG-16	Cross Entropy Loss	random color, random rotation, random flip	IoU, Precision, Recall, F1 Score
[74]	RGB	Raster	WHU Aerial Building dataset	U-Net	Custom	Cross Entropy Loss	random scale, random color, random rotation, random flip	Precision, Recall, F1 Score

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[66]	RGB	Raster	Custom	D-LinkNet	Custom	Custom Loss	random crop, random flip, random rotate	IoU, Precision, Recall, F1 Score
[101]	RGB,NIR	Raster	Custom	LSTM	Custom	no info	no info	IoU, Precision, Recall, F1 Score, Overall Accuracy
[51]	RGB	Raster	WHU Aerial Building dataset	U2-Net	Residual U-Blocks	Multiclass Cross Entropy Loss	random flip, random rotation	F1 Score, mIoU
[99]	RGB	Raster	WHU Aerial Building dataset	Custom U-Net (EPUNet)	ResNet-34	Custom Loss	random scale, random rotation, random flip	Overall Accuracy, Precision, Recall, F1 Score, IoU
[83]	RGB	Raster	INRIA Aerial Image Labeling Dataset	U-Net	Custom	Weighted Loss Network, Modified Cross Entropy Loss	no info	Overall Accuracy, mIoU, Kappa
[67]	RGB	Raster	DREAM-B, Massachusets Buildings dataset	U-Net	Custom (ESPC module)	Categorical Cross Entropy loss	random color, random rotation, random flip	Overall Accuracy, Kappa, Precision, Recall, F1 Score, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

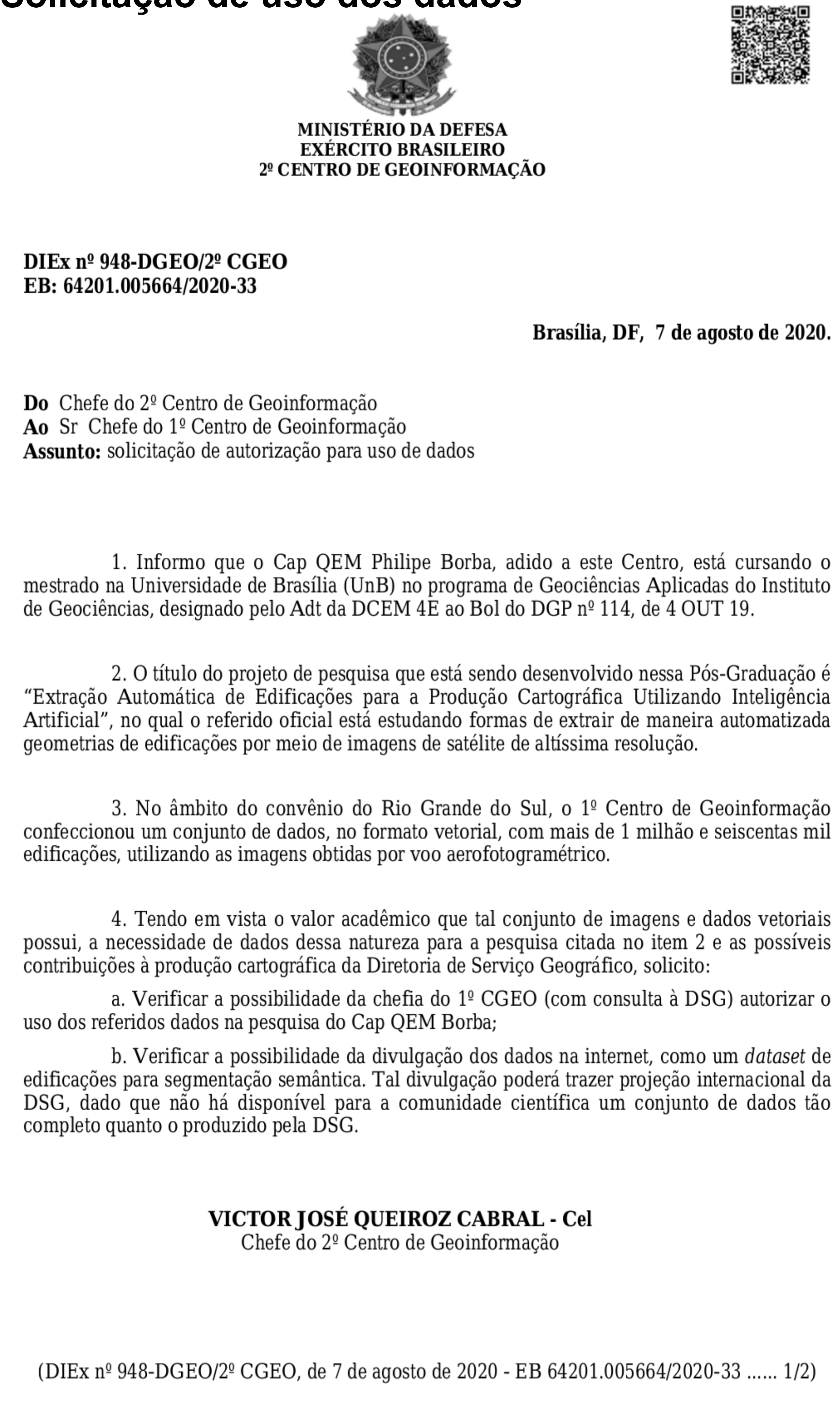
Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[55]	RGB	Raster	Custom	U-Net	Custom	Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score, IoU
[82]	RGB	Raster	Custom	DLinkNet, U-Net	ResNet-34	Joint loss (reconstruction loss, adversarial loss, content loss, total variation regularization loss)	no info	IoU, Precision, Recall, F1 Score, Kappa
[75]	RGB	Raster	WHU Aerial Building dataset, Massachusets Buildings dataset	MHA-Net	Custom	Binary Cross Entropy	random color, random rotation, random flip	Precision, Recall, F1 Score, IoU

Continued on next page

Table 19: Building footprint extraction papers full analysis.

Paper	Input Type	Output Type	Dataset	Architecture	Backbone	Loss	Data Augmentation	Evaluation Metrics
[62]	RGB, NDVI, nDSM	Raster	Massachusetts Buildings dataset, WHU Aerial Building dataset	FPN	ResNet-50	Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score
[98]	RGB	Raster	ISPRS Potsdam, AISD Dataset	NFSNet	ResNet-18	Cross Entropy Loss	NaN	Overall Accuracy, Precision, Recall, F1 Score, mIoU
[91]	RGB	Raster	WHU Aerial Building dataset, Massachusets Buildings dataset	Custom	Custom	Binary Cross Entropy Loss	random rotation	Precision, Recall, F1 Score, IoU
[92]	RGB	Raster	WHU Aerial Building dataset	HRNetV2	Custom	Binary Cross Entropy Loss	no info	Overall Accuracy, Precision, Recall, F1 Score

B Solicitação de uso dos dados



MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
2º CENTRO DE GEOINFORMAÇÃO

DIEx nº 948-DGEO/2º CGEO
EB: 64201.005664/2020-33

Brasília, DF, 7 de agosto de 2020.

Do Chefe do 2º Centro de Geoinformação
Ao Sr Chefe do 1º Centro de Geoinformação
Assunto: solicitação de autorização para uso de dados

1. Informo que o Cap QEM Philipe Borba, adido a este Centro, está cursando o mestrado na Universidade de Brasília (UnB) no programa de Geociências Aplicadas do Instituto de Geociências, designado pelo Adt da DCEM 4E ao Bol do DGP nº 114, de 4 OUT 19.

2. O título do projeto de pesquisa que está sendo desenvolvido nessa Pós-Graduação é "Extração Automática de Edificações para a Produção Cartográfica Utilizando Inteligência Artificial", no qual o referido oficial está estudando formas de extrair de maneira automatizada geometrias de edificações por meio de imagens de satélite de altíssima resolução.

3. No âmbito do convênio do Rio Grande do Sul, o 1º Centro de Geoinformação confeccionou um conjunto de dados, no formato vetorial, com mais de 1 milhão e seiscentas mil edificações, utilizando as imagens obtidas por voo aerofotogramétrico.

4. Tendo em vista o valor acadêmico que tal conjunto de imagens e dados vetoriais possui, a necessidade de dados dessa natureza para a pesquisa citada no item 2 e as possíveis contribuições à produção cartográfica da Diretoria de Serviço Geográfico, solicito:

a. Verificar a possibilidade da chefia do 1º CGEO (com consulta à DSG) autorizar o uso dos referidos dados na pesquisa do Cap QEM Borba;

b. Verificar a possibilidade da divulgação dos dados na internet, como um *dataset* de edificações para segmentação semântica. Tal divulgação poderá trazer projeção internacional da DSG, dado que não há disponível para a comunidade científica um conjunto de dados tão completo quanto o produzido pela DSG.

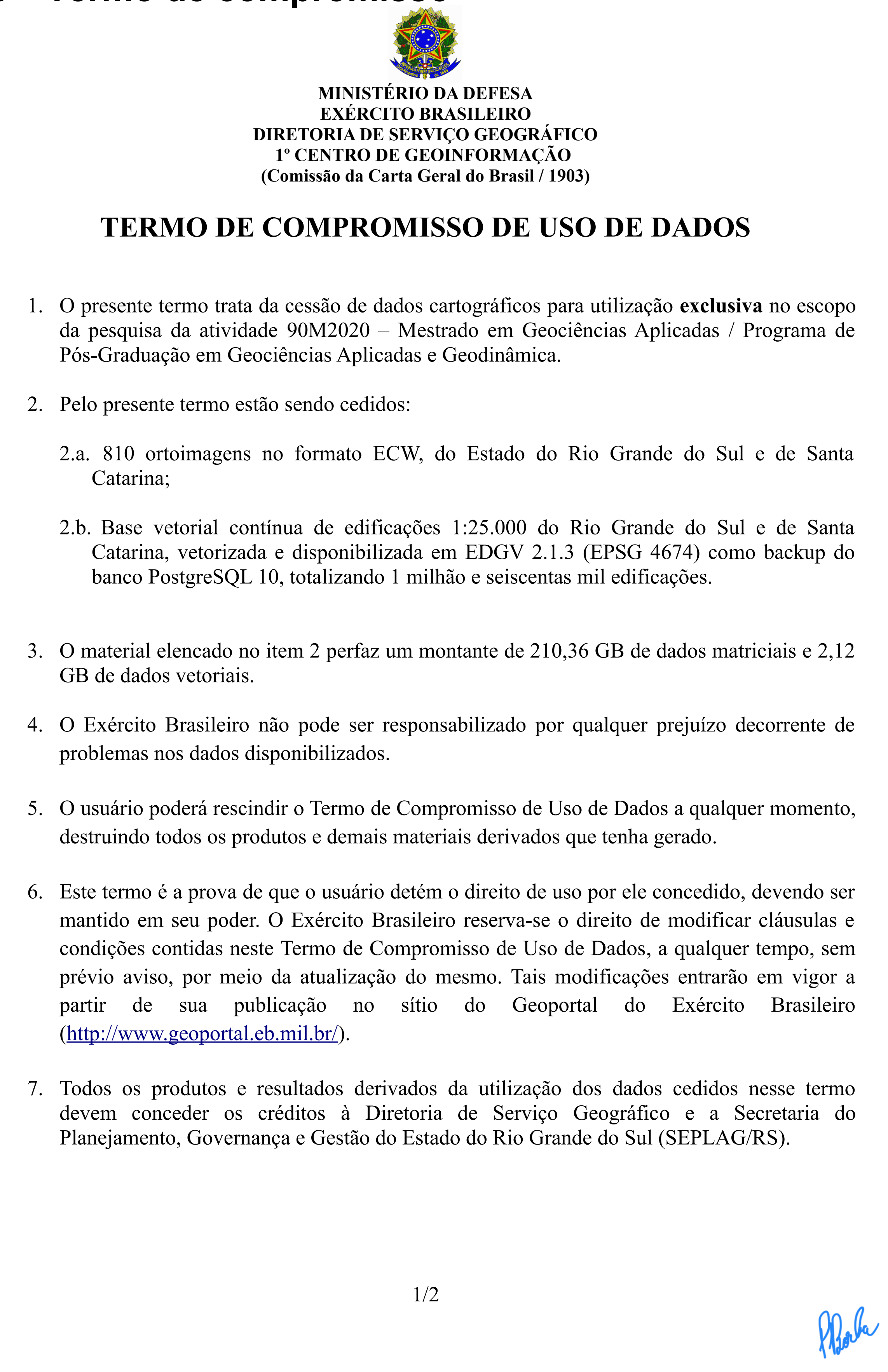
VICTOR JOSÉ QUEIROZ CABRAL - Cel
Chefe do 2º Centro de Geoinformação

(DIEx nº 948-DGEO/2º CGEO, de 7 de agosto de 2020 - EB 64201.005664/2020-33 ...... 1/2)

"INTENDÊNCIA: SOLDADO DO ACANTO,
UM SÉCULO DE EXCELÊNCIA NA LOGÍSTICA MILITAR TERRESTRE"

(DIEx nº 948-DGEO/2º CGEO, de 7 de agosto de 2020 - EB 64201.005664/2020-33 ...... 2/2)

C Termo de compromisso



MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DIRETORIA DE SERVIÇO GEOGRÁFICO
1º CENTRO DE GEOINFORMAÇÃO
(Comissão da Carta Geral do Brasil / 1903)

TERMO DE COMPROMISSO DE USO DE DADOS

1. O presente termo trata da cessão de dados cartográficos para utilização **exclusiva** no escopo da pesquisa da atividade 90M2020 – Mestrado em Geociências Aplicadas / Programa de Pós-Graduação em Geociências Aplicadas e Geodinâmica.

2. Pelo presente termo estão sendo cedidos:

   2.a. 810 ortoimagens no formato ECW, do Estado do Rio Grande do Sul e de Santa Catarina;

   2.b. Base vetorial contínua de edificações 1:25.000 do Rio Grande do Sul e de Santa Catarina, vetorizada e disponibilizada em EDGV 2.1.3 (EPSG 4674) como backup do banco PostgreSQL 10, totalizando 1 milhão e seiscentas mil edificações.

3. O material elencado no item 2 perfaz um montante de 210,36 GB de dados matriciais e 2,12 GB de dados vetoriais.

4. O Exército Brasileiro não pode ser responsabilizado por qualquer prejuízo decorrente de problemas nos dados disponibilizados.

5. O usuário poderá rescindir o Termo de Compromisso de Uso de Dados a qualquer momento, destruindo todos os produtos e demais materiais derivados que tenha gerado.

6. Este termo é a prova de que o usuário detém o direito de uso por ele concedido, devendo ser mantido em seu poder. O Exército Brasileiro reserva-se o direito de modificar cláusulas e condições contidas neste Termo de Compromisso de Uso de Dados, a qualquer tempo, sem prévio aviso, por meio da atualização do mesmo. Tais modificações entrarão em vigor a partir de sua publicação no sítio do Geoportal do Exército Brasileiro (http://www.geoportal.eb.mil.br/).

7. Todos os produtos e resultados derivados da utilização dos dados cedidos nesse termo devem conceder os créditos à Diretoria de Serviço Geográfico e a Secretaria do Planejamento, Governança e Gestão do Estado do Rio Grande do Sul (SEPLAG/RS).

1/2

8. Este Termo de Compromisso de Uso de Dados é feito em concordância com as leis do Brasil.

Porto Alegre-RS, 28 de setembro de 2020.

Philipe Borba

Philipe Borba – Cap QEM
idt: 010300117-8