Universidade de São Paulo
Instituto de Geociências

Segmentação automática de cicatrizes de deslizamento de terra em imagens de sensores remotos utilizando aprendizagem profunda de máquina (*Deep Learning*)

Lucas Pedrosa Soares

Orientador: Prof. Dr. Carlos Henrique Grohmann de Carvalho

Dissertação de Mestrado
Programa de Pós-Graduação em Recursos Minerais e Hidrogeologia

São Paulo
2022

Universidade de São Paulo
Instituto de Geociências

Segmentação automática de cicatrizes de deslizamento de terra em imagens de sensores remotos utilizando aprendizagem profunda de máquina (*Deep Learning*)

Lucas Pedrosa Soares

Dissertação de mestrado apresentada ao programa de Pós-Graduação em Recursos Minerais e Hidrogeologia do Instituto de Geociências para obtenção do título de mestre em Ciências.

Orientador: Prof. Dr. Carlos Henrique Grohmann de Carvalho

São Paulo
2022

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE GEOCIÊNCIAS

SEGMENTAÇÃO AUTOMÁTICA DE CICATRIZES DE DESLIZAMENTO DE TERRA EM IMAGENS DE SENSORES REMOTOS UTILIZANDO APRENDIZAGEM PROFUNDA DE MÁQUINA (DEEP LEARNING)

LUCAS PEDROSA SOARES

Orientador: Prof. Dr. Carlos Henrique Grohmann de Carvalho

Dissertação de Mestrado

Nº 888

COMISSÃO JULGADORA

Dr. Carlos Henrique Grohmann de Carvalho

Dra. Leila Maria Garcia Fonseca

Dr. Rogério Galante Negri

SÃO PAULO
2022

AGRADECIMENTOS

Agradeço a minha família e amigos por todo apoio prestado durante todo esse período (difícil) do mestrado. Enfrentar uma pandemia e diversas incertezas em meio a todo esse processo foi extremamente desafiador. Manter as forças para seguir em frente só foi possível pelo apoio incondicional e amor que recebi da minha família. O tempo é professor para nos ensinar o poder de ter pessoas que te amam ao lado, cada vez fica mais claro o quão sortudo eu sou de tê-los à minha volta.

A outra sorte que tive foi de encontrar no SPAMLab um lar onde fiz bons amigos que me ajudaram muito em todo esse processo. Deixo aqui meus agradecimentos especiais ao Guano por ser sempre preciso nas orientações, solícito e aberto as ideias propostas durante esse projeto. A Helen, Rafael, Guilherme, Alynne e Gabriel deixo um muito obrigado por todas conversas, campos e reflexões que tivemos no tempo que foi possível nos encontrarmos no laboratório. Infelizmente a pandemia acabou limitando um pouco nossa interação. Porém, as pesquisas seguem a todo vapor. Tenho muita admiração por todos vocês.

" Nenhuma Sociedade que esquece a arte de questionar pode esperar encontrar respostas para os problemas que a afligem. "

Zygmunt Bauman

RESUMO

O número de casos de deslizamento de terra tem aumentado em todo o mundo em função de processos de urbanização e ocupação do território, além da maior frequência de eventos climáticos extremos. Os estudos até hoje realizados mostram que as redes de convolução têm obtido os melhores resultados para segmentação de cicatrizes de deslizamento de terra em imagens de sensores remotos. Porém, ainda são necessárias mais pesquisas nesta área, já que os métodos utilizados para o pré-processamento das imagens, arquiteturas e calibração dos parâmetros do modelo ainda representam um grande desafio para se obter modelos acurados. Este estudo teve como objetivo principal automatizar o processo de segmentação de cicatrizes de deslizamento de terra em imagens obtidas por sensores remotos por meio de técnicas de *deep learning* utilizando redes neurais convolucionais (RNC). A região Serrana do Rio de Janeiro, palco do maior desastre natural do Brasil, ocorrido em janeiro de 2011, e a região de Rolante, no Rio Grande do Sul, foram utilizadas como área de estudo. As imagens utilizadas para treinar os modelos foram obtidas pelo satélite *RapidEye*. A pesquisa avaliou a capacidade de generalização dos modelos em áreas distintas das de treinamento, além disso, concentrou-se no treinamento da rede U-Net, utilizando diferentes métodos de amostragem, dimensões de *input* e *datasets*, para avaliar como isto impacta na acurácia da segmentação dos deslizamentos de terra. Os resultados sugerem que os modelos treinados com imagens de maior dimensão (128 x 128 e 256 x 256 pixels) tendem a ser mais acurados em áreas similares a área de treinamento, enquanto os modelos treinados com imagens de menor dimensão (32 x 32 e 64x64 pixels) tendem a possuir uma capacidade de generalização melhor, consequentemente, obtiveram os melhores resultados em áreas que diferem das áreas de treino. O pós-processamento dos resultados com operações morfológicas é eficiente para melhorar a precisão dos resultados.

Palavras-Chave: *Deep Learning*, Redes Neurais de Convolução (RNC), Mapeamento de deslizamentos de terra.

ABSTRACT

The global number of landslides cases has increased due to greater urbanization and land use and a higher frequency of extreme climatic events. Since these events have substantial social and economic impacts on Brazil and worldwide, they are considered one obstacle to sustainable development, according to the United Nations (UN). Studies show that convolution neural networks (CNN) achieve the best results on landslide segmentation in aerial imagery. However, choosing the best network architecture, preparing the dataset, and tuning the hyperparameters is still challenging. This study's main objective is to use deep learning with convolutional neural networks (CNN) to automate the segmentation of landslide scars on remote sensing imagery. The study area is located in the mountainous range of Rio de Janeiro state, where, in January 2011, the worst Brazilian Natural Disaster occurred. In addition, the area located in Rolante, in the Rio Grande do Sul state, was also used as a study area. The images used to train the models were acquired from the RapidEye satellite. The research focused on evaluating the generalization capacity of the models em areas that differ from the training area. Moreover, the study also evaluated how different sampling methods, patch sizes, and datasets impact the overall accuracy of U-Net on landslide segmentation. The results suggest that the models trained with the bigger patch sizes (128x128 and 256x256 pixels) tend to achieve better accuracy in areas similar to the training areas, while models trained with the smaller patch sizes (32x32 pixels and 64x64 pixels) achieved better generalization results in areas that are different from the training images. Postprocessing the results were efficient to increase the precision of the models.

***Keywords**: Deep Learning, Convolution Neural Networks (CNN), landslide mapping.*

LISTA DE FIGURAS

SUMÁRIO

1 INTRODUÇÃO E JUSTIFICATIVA

O impacto dos desastres naturais se tornou mais severo nas últimas décadas, elevando os custos para a economia dos países e o número de pessoas afetadas. O alto crescimento populacional aliado a processos de urbanização e ocupação desordenada favorecem a ocupação de áreas de risco e tem grande relevância para o aumento no número de vítimas (KOBIYAMA et al., 2006). Além disso, os eventos climáticos estão mais frequentes e extremos, favorecendo uma ocorrência mais constante destes desastres.

De acordo com o Marco de Ação Sendai para a Redução de Risco a Desastres 2015-2030 (UNISDR, 2015), emitido pela Organização das Nações Unidas (ONU), os desastres são uma das principais barreiras ao desenvolvimento sustentável, já que, entre 2008 e 2012, os prejuízos econômicos ultrapassaram 1,3 trilhões de dólares, além disso, mais de 700 mil pessoas perderam a vida, 1,4 milhões de pessoas ficaram feridas e 23 milhões de pessoas desabrigadas.

No Brasil, os deslizamentos de terra, inundações e enxurradas são responsáveis por grande parte dos prejuízos gerados por desastres. Segundo o *Emergency Events Database* (EM-DAT, 2018), no período de 2001 a 2017, sete milhões de pessoas foram afetadas e foram registrados aproximadamente 2.500 óbitos e os prejuízos ultrapassaram R$ 19 bilhões. Neste cenário, destacam-se os dois maiores desastres gerados por movimentos de massa no país: os eventos ocorridos no ano de 2008 no vale do Itajaí, em Santa Catarina, onde mais de 1.5 milhões de pessoas sofreram danos e 120 óbitos foram registrados e o evento de 2011 na região serrana do Rio de Janeiro, que afetou diretamente mais de 1 milhão de pessoas e casou 1.100 óbitos.

Deslizamentos de terra são caracterizados por movimentos de solo, rocha e/ou vegetação ao longo da vertente sob influência direta da gravidade. Fatores como a água e o gelo podem contribuir neste processo, uma vez que diminuem a resistência dos materiais e afetam seu comportamento reológico (TOMINAGA; SANTORO; AMARAL, 2009). Além disso, as condições geomorfológicas, pedológicas, geológicas, climáticas e hidrológicas originais da área podem atuar como agentes predisponentes na deflagração de movimentos de massa. Sendo as chuvas inten-

sas, erosão, terremotos e tsunamis os principais fatores desencadeadores deste processo.

Os modelos de susceptibilidade e risco a deslizamentos dependem de mapas de inventário de cicatrizes de eventos anteriores para serem validados. Porém, estes mapas são escassos e muitas vezes inexistentes já que o processo de elaboração destes mapas, a depender das dimensões da área, é lento e não apresenta uma sistemática bem definida sendo, por isso, muitas vezes subjetivo.

Os mapas de inventário devem conter a localização, forma, limites, data de ocorrência, tipo e outras informações do movimento de massa (HERVÁS; BOBROWSKY, 2009) registradas em um banco de dados acessível em plataforma de sistema de informações geográficas (SIG). Estes mapas podem ser elaborados de maneira manual, semiautomática e automática. Os métodos manuais se baseiam na coleta de dados de campo e na interpretação das cicatrizes em imagens de sensoriamento remoto. Já os semiautomáticos e automáticos utilizam algoritmos de classificação baseados em objetos e em pixels (DIAS et al., 2021).

Os métodos baseados em pixels buscam classificar individualmente os pixels de uma imagem com base em sua informação espectral. Dentre os métodos utilizados, destacam-se a classificação supervisionada e não supervisionada, detecção de mudanças e segmentação baseada em um limite (*threshold*). Os métodos baseados em pixels, apesar de serem os mais comuns, muitas vezes não apresentam boa acurácia na segmentação de cicatrizes de deslizamentos de terra, pois não consideram informações geométricas e contextuais da imagem o que aumenta o ruído dos resultados (STUMPF; KERLE, 2011; BLASCHKE et al., 2014; ZHONG et al., 2019; PRAKASH; MANCONI; LOEW, 2020). Já os métodos baseados em objetos combinam informações espaciais, espectrais, morfológicas em uma abordagem semi-automática, que depende da experiência do especialista, para classificar um conjunto de pixels de uma imagem (BLASCHKE, 2010). Este método reduz o ruído da classificação e permite que as características do deslizamento sejam extraídas com maior acurácia.

Nos últimos anos, o *deep learning* tem se tornado o estado da arte em problemas de visão computacional, sendo, por isso, cada vez mais utilizado em sensoriamento remoto para aplicações de segmentação semântica, detecção de objetos e

classificação de cenas em imagens de alta resolução (GHORBANZADEH; BLAS-CHKE et al., 2019; PENG; ZHANG; GUAN, 2019; ZHU et al., 2017; LONG; SHE-LHAMER; DARRELL, 2015; RADOVIC; ADARKWA; WANG, 2017). Porém, existem poucos trabalhos que utilizam estas técnicas para a detecção de deslizamentos de terra (ZHONG et al., 2019).

Os modelos de *deep learning* são avaliados com base na análise dos valores de verdadeiros positivos (VP), falso positivos (FP) e falsos negativos (FN). Com estes valores é possível calcular as métricas de precisão, *recall*, *f1-score* e índice de Jaccard que permitem uma análise mais detalhada da performance do modelo.

Ding et al. (2016) utilizaram redes neurais convolucionais (RNC) para detectar automaticamente deslizamentos de terra em imagens Gaofen-1 (GF-1) com quatro bandas espectrais e oito metros de resolução espacial. O método obteve 67% de acurácia com a taxa de detecção de 72.5% e taxa de falsos positivos de 10.2%.

Chen et al. (2018) utilizaram redes neurais convolucionais profundas em imagens bi-temporais para identificar áreas que passaram por mudanças abruptas e, posteriormente, realizaram uma análise espaço-temporal e avaliaram mudanças em um modelo digital de elevação para identificar os deslizamentos. O método obteve acurácia superior a 61.1% nas áreas avaliadas.

Ghorbanzadeh, Blaschke et al. (2019) utilizaram imagens *RapidEye* e um modelo digital de elevação (MDE), ambos com 5 metros de resolução espacial, para comparar métodos de aprendizado de máquina (redes neurais artificiais, *support vector machines* e *random forest*) com modelos de aprendizado profundo de máquina para segmentar deslizamentos de terra no distrito de Rasuwa, Nepal. Os resultados obtidos mostraram que a rede de convolução apresentou o melhor resultado, obtendo um índice de Jaccard de 78,26%. Ainda, as informações obtidas do MDE ajudaram a discriminar as áreas urbanas de deslizamentos de terra, porém, não melhoraram a acurácia geral do modelo.

Sameen e Pradhan (2019) compararam a performance de redes Residuais (ResNets) treinadas com informações topográficas que foram fusionadas à informações espectrais utilizando redes de convolução com ResNets que foram treinadas com dados topográficos adicionados como bandas extras em imagens espectrais. O resultado mostra que os modelos que utilizaram RNCs para fusionar os dados

obtiveram *f1-score* superior em 13% e índice de Jaccard em 12,96%.

Yu, Chen e Xu (2020) utilizaram índices de vegetação e um MDE para obter áreas com potencial à ocorrência de deslizamentos de terra no Nepal. Posteriormente, estes dados serviram para treinar a rede *Pyramid Scene Parsing Network* (PspNet) na segmentação de cicatrizes em imagens Landsat. O método obteve 65% de *recall* e 55,35% de acurácia.

Prakash, Manconi e Loew (2020) utilizaram a arquitetura de rede U-Net em conjunto com a ResNet34 e compararam esses resultados com métodos tradicionais de classificação de pixel e métodos baseados em objetos. Os modelos foram treinados com imagens do Sentinel-2 e com um MDE gerado por lidar. A rede U-Net + ResNet34 obteve os melhores resultados com um coeficiente de correlação de Matthews de 0,495 e taxa de detecção de 0,72.

Liu et al. (2020) utilizaram a rede U-Net com algumas modificações para segmentar escorregamentos de terra gerados após terremotos. As imagens utilizadas no treinamento do modelo consistem das bandas RGB, MDE, declividade, aspecto e possuem resolução de 0,14 metros. Os resultados mostram que o método utilizado obteve acurácia de 91,3%.

Shi et al. (2020) combinaram as técnicas de detecção de mudanças e redes neurais convolucionais para detectar escorregamentos de terra nas regiões de Lantau Island e Sharp Peak em Hong Kong. A técnica obteve uma acurácia de 80%.

Qi et al. (2020) utilizaram a rede ResU-Net para mapear escorregamentos de terra de escala regional automaticamente. Os modelos foram treinados com as bandas RGB das imagens *GeoEye-1* com 0,5 metros de resolução. Os resultados mostram que a ResU-Net obteve resultados superiores, em 9%, em relação a U-Net.

Ghorbanzadeh, Meena et al. (2020) utilizaram uma rede neural convolucional e a técnica de *Dempster-Shafer* (D-S) para avaliar qual a relevância de dados óticos e topográficos na predição dos modelos. As imagens utilizadas foram do sensor *Planet-Scope* e ALOS. Os resultados mostram que utilizar dados óticos em conjunto com dados de topografia, em específico o *slope*, auxiliam a diferenciar os escorregamentos de outros elementos com resposta espectral similar.

Yi e Zhang (2020) desenvolveram a arquitetura de rede convolucional conhecida como *LandsNet*, que permite que o modelo aprenda características dos es-

corregamentos com diferentes dimensões espaciais. O modelo obteve f1-*score* de 86,89% e foi superior em 7% e 8% em relação a ResU-Net e a U-Net, respectivamente.

Meena et al. (2021) utilizaram imagens da *Planet-Scope* e fatores topográficos (declividade) parar segmentar, com redes convolucionais, escorregamentos de terra gerados devido ao sistema de monções, na Índia. A acurácia média aumentou de 65.5% para 78% quando os dados topográficos foram utilizados.

Prakash, Manconi e Loew (2021) avaliaram a capacidade de generalização da ResU-Net para segmentar escorregamentos de terra em diferentes localidades do mundo. O método combina imagens de escorregamentos de diferentes áreas de uma maneira progressiva para melhorar a capacidade de generalização destes modelos. Os resultados mostram que a técnica é útil para segmentar escorregamentos em áreas que não possuem inventário local.

Ghorbanzadeh, Crivellari et al. (2021) utilizaram as imagens Sentinel-2 e ALOS para segmentar escorregamentos em três áreas distintas. O estudo foi o primeiro a utilizar imagens disponibilizadas gratuitamente com a técnica de deep learning. O maior valor de *f1-score* obtido foi de 73.32%.

Ghorbanzadeh, Shahabi et al. (2022) integraram a técnica de *deep learning* com a análise baseada em objetos (OBIA) para detectar os escorregamentos. Os modelos foram treinados com imagens Sentinel-2. A metodologia consistiu em combinar o resultado obtido no modelo de *deep learning* com a imagem original e aplicar a técnica de OBIA. O resultado mostra que esse modelo híbrido conseguiu valores que são 22% superiores em relação a modelos que utilizarem as técnicas individualmente.

Os trabalhos até aqui realizados evidenciam que as redes de convolução apresentam uma maior acurácia para segmentar os deslizamentos de terra, quando comparadas a outros modelos de aprendizado de máquina. Além disso, nota-se que é comum um desbalanceamento entre a classe de interesse (deslizamento de terra) e a classe negativa (*background*) nas imagens utilizadas nos treinamentos do modelo, dificultando este processo, já que são necessárias metodologias que permitam reduzir este desbalanceamento. Em conjunto a isto, os resultados destes trabalhos ainda indicam que as redes de convolução, em problemas de aprendizado super-

visionado, podem aprender representações dos dados sem que sejam necessárias técnicas complexas de pré-processamento. Porém, ajustar os dados de *input*, configurar os parâmetros e arquiteturas do modelo ainda é um desafio, sendo necessário mais estudos que abordem tais temáticas.

Neste cenário, estudos que avaliem diferentes métodos de amostragem, dimensões das imagens de treinamento, configurações e arquiteturas de rede são relevantes, uma vez que existem poucos trabalhos que tiveram tal finalidade. Assim, este trabalho teve como objetivo principal utilizar as redes neurais convolucionais para segmentar cicatrizes de deslizamento de terra em imagens de sensores remotos de maneira automática e avaliar o impacto de diferentes arquiteturas de rede, métodos de amostragem e dimensões das imagens de treinamento na acurácia do modelo. Além disso, um dos focos do estudo foi também entender a capacidade de generalização desses modelos. Para isso, foram treinados redes totalmente convolucionais (*fully convolutional network*), U-Net, utilizando imagens *RapidEye* de Nova Friburgo e Teresópolis, localizadas na região Serrana do Rio de Janeiro (RJ), e de Rolante, no Rio Grande do Sul (RS).

2 OBJETIVOS

2.1 Objetivo Geral

- Segmentar automaticamente cicatrizes de deslizamento de terra utilizando redes neurais convolucionais.

2.2 Objetivos Específicos

- Avaliar diferentes arquiteturas e parâmetros de rede;

- Avaliar diferentes formas de amostragem e dimensões de input;

- Avaliar o impacto das técnicas *data augmentation* e informações topográficas na acurácia do modelo;

- Avaliar a capacidade de generalização dos modelos;

- Avaliar técncias de pós-processamento dos resultados.

3 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado na forma de artigos, portanto, sua estrutura contém três seções principais, a saber, fundamentação teórica (seção 4), resultados (seção 5) e conclusões (seção 6). Na seção de fundamentação teórica serão apresentados aspectos referentes aos deslizamentos de terra, sensoriamento remoto, *deep learning* e operações morfológicas de pós-processamento. A seção de resultados apresenta um resumo, em português, dos dois artigos desenvolvidos durante esta pesquisa e a referência para outros dois trabalhos que foram desenvolvidos em coautoria com pesquisadores internacionais. Os textos completos, em inglês, estão na seção de apêndices (seção 7). A última seção corresponde as conclusões que foram obtidas durante a pesquisa.

4 FUNDAMENTAÇÃO TEÓRICA

4.1 Movimentos de Massa

Os movimentos de massa são caracterizados como o deslocamento de qualquer material terroso ou rochoso sobre a influência direta da gravidade. Diversos autores apresentam sistemáticas para classificar estes movimentos (VARNES, 1978; AUGUSTO FILHO, 1992; USGS, 2004). A classificação de Augusto Filho (1992) foi elaborada para caracterizar escorregamentos para o meio tropical e subtropical úmido, tipo climático que é predominante na região Serrana do Rio de Janeiro e na Serra do Mar (DIAS, 2019). Por isso, essa foi adotada neste trabalho. O autor divide os movimentos de massa em quatro tipologias: rastejos, escorregamentos, quedas e corridas. A caracterização dos movimentos é feita com base na geometria, mecanismo de movimentação, natureza, estado do material mobilizado e comportamento ao longo do tempo.

Os rastejos (figura 1-A) são movimentos lentos e graduais que ocorrem em terrenos com alta declividade. Feições como degraus de abatimentos em encostas, trincas em paredes e árvores inclinadas podem indicar este tipo de movimento.

As corridas (figura 1-B) ocorrem em velocidades elevadas devido a chuvas intensas que mobilizam uma grande quantidade de material ao longo das drenagens. Estes movimentos transportam um grande volume de material, por isso, têm alto poder destrutivo.

Os escorregamentos (figura 1-C) apresentam velocidades médias à altas com planos de ruptura bem definidos e variável volume de material transportado. O seu mecanismo de movimentação está associados ao tipo de relevo, solo e rocha, sendo divididos em três tipos: cunha, circulares e planares.

Os escorregamentos circulares possuem superfície de ruptura curva que deslocam grande quantidade de material de forma rotacional. Áreas com rochas muito fraturadas e com solos bem desenvolvidos e homogêneos são mais susceptíveis a sua ocorrência. Os escorregamentos planares ou rasos apresentam superfície de ruptura planar que, no geral, coincide com as descontinuidades do material (juntas, falhas, limites litológicos). São processos que ocorrem em porções superficiais da encosta. Os escorregamentos em cunha estão associados com solos e rochas com

dois planos de fraqueza.

No presente trabalho, os termos "deslizamento de terra" e, em inglês, "*lands-lides*", serão utilizados como sinônimos dos três tipo de processos de escorregamento (rotacional, planar e cunha).

As quedas (figura 1-D) estão associadas a movimentação de alta velocidade de materiais rochosos ao longo de uma encosta. Os principais tipos de movimentação agrupados nesta categoria são: tombamentos de blocos, rolamento de blocos, queda de blocos e desplacamento de lascas e blocos de rocha.
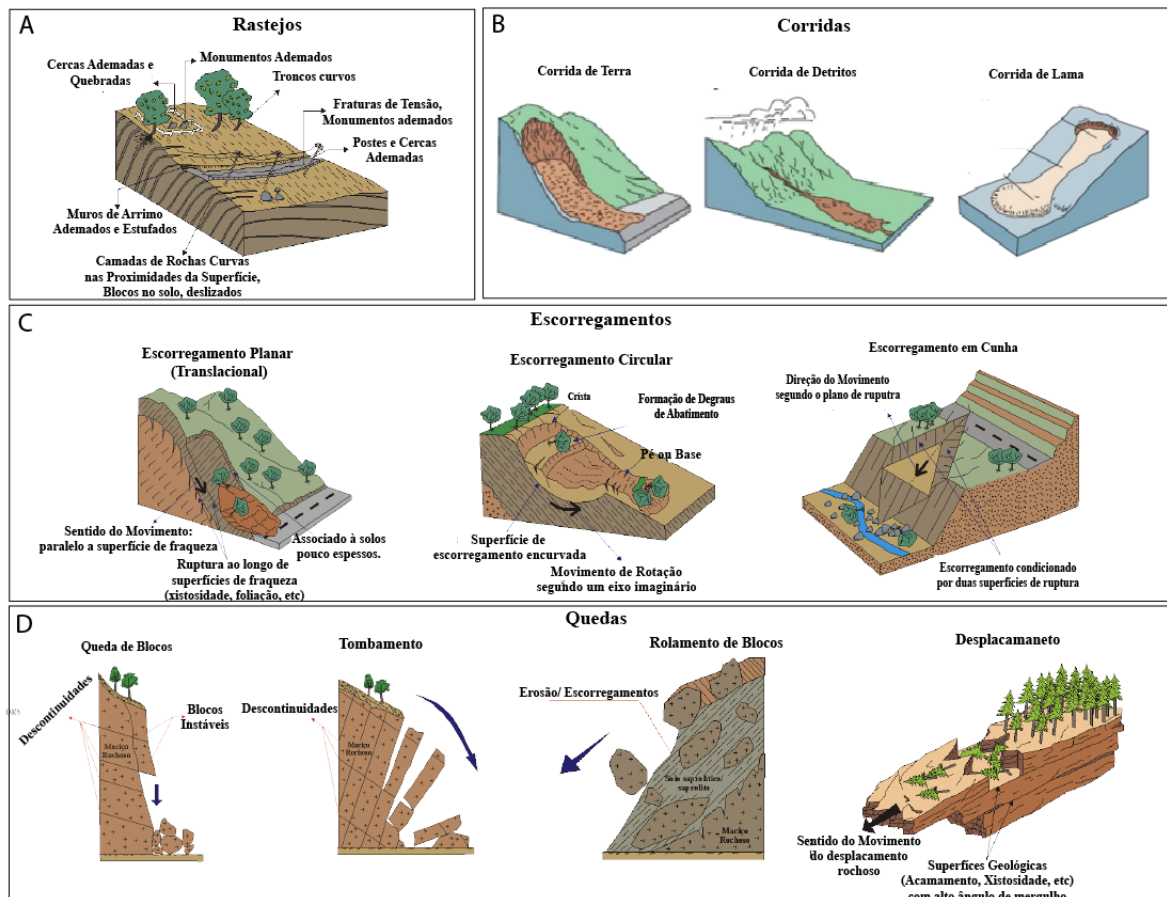
Figura 1: Tipos de movimento de massa.

Fonte: Adaptado de rc.unesp.br/igce/aplicada/ead/interacao/inter09.html, acessado em 01/02/2022.

4.2 Sensoriamento Remoto

O sensoriamento remoto é uma ciência que visa obter imagens da superfície terrestre por meio da detecção e medição quantitativa das respostas das interações da radiação eletromagnética (REM) com os materiais terrestres (MENESES; ALMEIDA, 2012). Qualquer objeto com temperatura acima de 0 Kelvin emite REM, porém, para que uma imagem seja gerada, é necessário que a fonte desta radiação possua uma alta taxa de transferência de energia. Com isso, o Sol e a Terra são as principais fontes naturais de REM utilizadas em sensoriamento remoto (figura 2). O Sol emite um espectro contínuo que vai de comprimentos de ondas menores (ultravioleta) a comprimentos maiores (infravermelho próximo e infravermelho de ondas curtas). Por outro lado, a Terra emite comprimentos de ondas maiores que do sol, na faixa do infravermelho. Existem alguns comprimentos de onda, como as microondas, que não podem ser emitidas por estas fontes, sendo por isso necessário o uso de fontes artificiais.
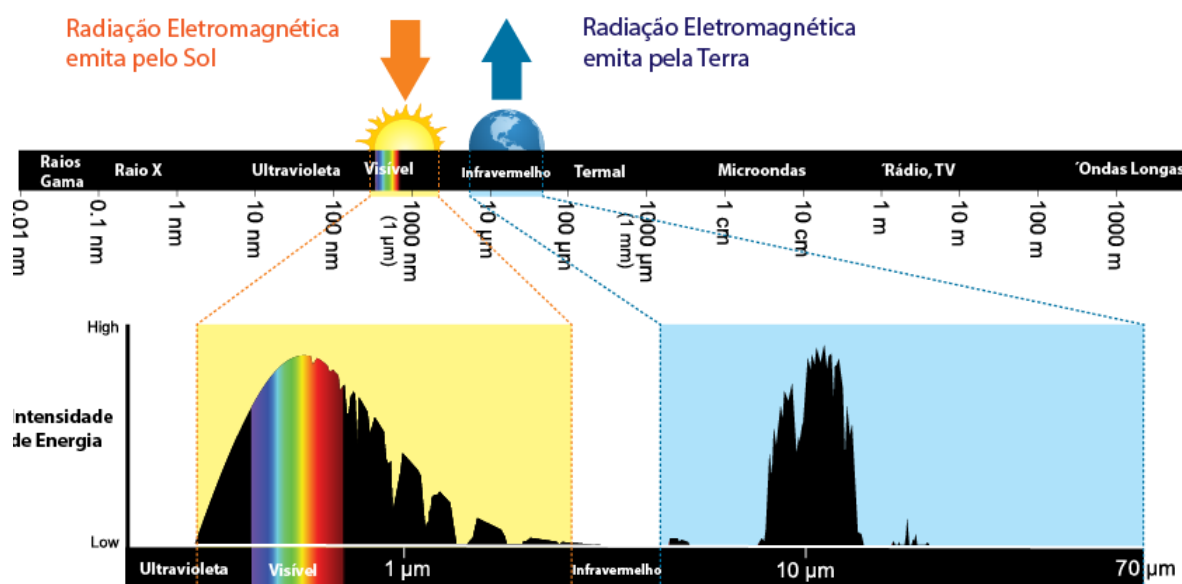
Figura 2: Espectro eletromagnético emitido pelo Sol e Terra.

Fonte: Modificado de https://www.weather.gov/jetstream/absorb, acessado em 23/06/2020.

A radiação eletromagnética refletida pelos alvos terrestres deve ser captada por sensores instalados em plataformas de aquisição (satélites, aeronaves, bases de observação, etc), para que seja gerada uma imagem de sensoriamento remoto (figura 3) . Para isso, a radiação eletromagnética que parte de uma fonte (A), aqui

o sol será utilizado como exemplo, deve atravessar a atmosfera (B) e chegar a um alvo na superfície terrestre (C) e, após interagir com este alvo, retornar a um sensor (D), atravessando novamente a atmosfera, porém agora no caminho ascendente. O sensor registra o valor de radiância, emitância e retroespalhamento dos alvos [1] em pixels cujos valores são discretizados em números digitais (MENESES; ALMEIDA, 2012). Posteriormente, existem estações terrestres (E) que registram o sinal digital enviado pelos satélites e o convertem em uma imagem digital, também conhecida como *raster*. Antes da disponibilização destas imagens (G), em geral, são realizadas correções atmosféricas, geométricas, radiométricas e de realce (F).
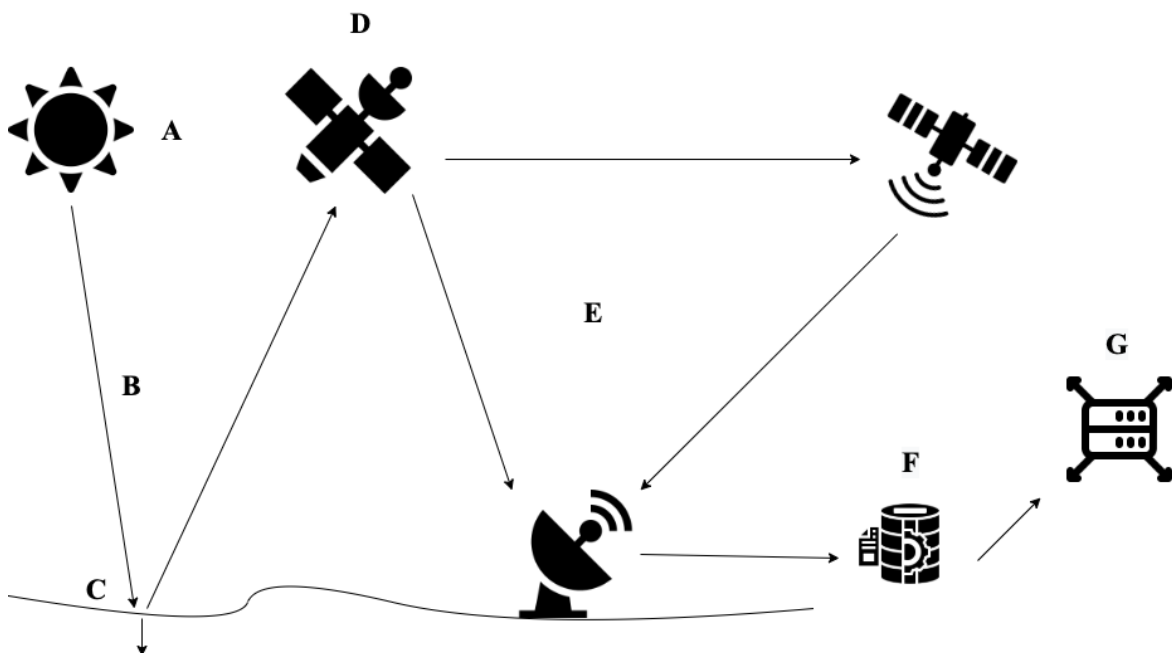
Figura 3: Processo de aquisição de uma imagem em sensoriamento remoto.

Fonte: elaborado pelo autor.

A intensidade da interação entre a REM e o alvo é definida por seu comprimento de onda. As interações no nível macroscópico ocorrem em maiores comprimentos de onda (com menor energia), que não geram interações em níveis atômicos. Com isso, as interações ocorrem apenas nos níveis superficiais, o que favorece a identificação de características morfológicas dos alvos como textura e rugosidade

---

[1]A radiância é a medida da densidade de fluxo radiante que se propaga em uma direção definida por um cone elementar naquela direção. A emitância mede a energia emitida por um corpo por segundo e unidade de área. O retroespalhamento é a reflexão dos sinais em sua direção de partida. (MENESES; ALMEIDA, 2012).

(JENSEN, 2009). Já os menores comprimentos de onda carregam energia suficiente para gerar transições nos níveis de energia dos átomos e moléculas dos alvos. Desta forma, uma vez que materiais de diferentes composições apresentam diferentes absortância e reflectância[2], a parte que é refletida desta interação carregará uma assinatura espectral que contém informações sobre a composição do objeto (MENESES; ALMEIDA; MELLO BAPTISTA, 2019).

A análise espectral dos materiais permite que, em sensoriamento remoto, sejam realizadas interpretações que levam em conta não apenas a morfologia do material, mas também sua reflectância. Porém, para a escolha da imagem adequada para realizar esta interpretação, é necessário avaliar a resolução espacial, espectral, radiométrica e temporal do sensor que será utilizado, já que estes fatores são limitantes a esta análise.

A resolução espacial se refere ao intervalo amostral, e por consequência, ao tamanho do menor objeto passível de ser identificado em uma imagem. Por definição, só é possível detectar um objeto quando seu tamanho é igual ou maior que o tamanho do elemento de resolução no terreno.

A resolução espectral define a largura das faixas do espectro eletromagnético que são adquiridas por um sensor. Um sensor tem uma melhor resolução espectral quando adquire um maior número de bandas situadas em diferentes regiões espectrais e com larguras estreitas, permitindo uma caracterização mais detalhada da reflectância de um determinado alvo.

A medida da intensidade de radiância da área de cada pixel define a resolução radiométrica. O número digital gerado no sensor é quantizado na forma de número de dígitos binários (*bits*). Imagens com uma maior quantidade de *bits*, por exemplo 16 (2^4) *bits*, registram maiores detalhes visuais em relação a imagens com menores resoluções radiométricas de 2, 4 e 8 (2^1, 2^2, 2^3) *bits*.

O tempo de revisita do sensor em determinada área define sua resolução temporal. Uma vez que existem variações sazonais que podem afetar a resposta espectral dos alvos, é fundamental avaliar a resolução temporal de uma imagem para se realizar uma caracterização espectral adequada. Além disso, para geração de

---

[2]A reflectância é calculada por meio da razão entre a radiância e a irradiância (quantidade de energia incidente naquela área). (MENESES; ALMEIDA; MELLO BAPTISTA, 2019)

séries temporais que avaliam mudanças ou evoluções na superfície terrestre, a resolução temporal tem um papel importante.

A resolução espacial do sensor é um dos fatores mais importantes na escolha de imagens para analisar e segmentar deslizamentos de terra, uma vez que só é possível identificar com clareza as feições geradas por estes eventos em imagens que apresentam resolução que seja similar a dimensão da cicatriz que é gerada no solo após um deslizamento. A resolução temporal também favorece a identificação dos deslizamentos já que é possível, por meio de análises temporais e subtrações entre as bandas de imagens obtidas em diferentes datas, identificar áreas que passaram por um maior processo de mudança. Porém, uma vez que os deslizamentos não apresentam respostas espectrais características que permita distingui-los de objetos com reflectância similar, como solos expostos, telhados e rios com elevada carga de fundo, comumente são utilizadas rotinas complexas de pré processamento de imagens e algoritmos de classificação na tentativa de acelerar e facilitar este processo identificação.

As técnicas de classificação digital de imagem visam implementar um processo de decisão que permita ao computador atribuir certo conjunto de pixels a uma determinada classe. A classificação baseada em pixel e baseada em objetos são os métodos de classificação de imagens mais utilizados. Recentemente, a técnica de *deep learning* (seção 4.3), utilizada neste trabalho, está sendo cada vez mais utilizada para classificar imagens em sensoriamento remoto.

4.2.1 Classificação Baseada em Pixels

A classificação baseada em pixels busca classificar individualmente cada pixel de uma imagem com base em seu padrão espectral. Este processo é o mais comum em sensoriamento remoto, podendo ser realizado de maneira supervisionada ou não supervisionada (JENSEN, 2009). Na classificação supervisionada o analista deve fornecer ao algoritmo amostras que contenham informações espectrais da classe de interesse. A partir destas informações, que são utilizadas como parâmetros de entrada no sistema de classificação, os algoritmos buscam classificar os pixels com comportamento espectral semelhante. Os algoritmos comumente utilizados são o de máxima verossimilhança (MaxVer), método do paralelepípedo e distância euclidiana, *Support Vector Machines (SVM)*, *AdaBoost*, *XgBoost* e *Deep*

Learning. Já na classificação não supervisionada o algoritmo busca agrupar pixels similares em grupos (*clusters*). Cada grupo é considerado como representativo da distribuição de probabilidade de uma determinada classe. Dentre as técnicas mais utilizadas em sensoriamento remoto destaca-se a *K-means*.

4.2.2 Classificação Orientada ao Objeto

A classificação baseada em objetos, *Geographic Object-Based Image Analysis* (GEOBIA), não classifica individualmente cada pixel, mas esta técnica agrupa pixels que apresentam similaridades espectrais, morfológicas e texturais em objetos primitivos. Para isso, é necessário que a imagem seja particionada em regiões (segmentação) que são posteriormente associadas a uma classe. É fundamental que a etapa de segmentação seja bem realizada, já que as características dos objetos ali delimitados serão utilizadas pelo algoritmo para segmentar objetos similares (BLASCHKE, 2010).

As técnicas de GEOBIA geralmente são indicadas em imagens de alta resolução espacial, já que estas imagens apresentam um nível maior de detalhe visual, o que aumenta a variabilidade espectral em um único alvo, gerando ruídos no processo de classificação por pixels. O agrupamento em objetos engloba esta variação de cada classe e, consequentemente, gera resultados menos ruidosos e com maior qualidade (ESMAEL et al., 2015).

4.3 *Deep Learning*

O *deep learning* é uma técnica de aprendizado de máquina em que se busca, por meio de redes neurais artificiais profundas, descobrir de maneira autônoma, isto é, sem programação explícita, regras e parâmetros a respeito de um conjunto de dados que possam fornecer uma representação adequada para o problema em questão (YAN; YOSHUA; GEOFFREY, 2015). O termo "aprendizado profundo" deriva da grande quantidade de camadas escondidas (*hidden layers*), localizadas entre a camada de entrada (*input*) e a camada de saída (*output*).

Os modelos podem ser treinados de maneira supervisionada, não supervisionada e reforçada. Neste trabalho, foi utilizado o método supervisionado, já que as imagens utilizadas no processo da rede foram previamente classificadas, servindo como um guia para o treinamento do modelo.

A escolha da arquitetura de rede está condicionada ao problema em questão. Uma vez que o objetivo deste trabalho é classificar os pixels da imagem nas classes escorregamentos de terra e *background*, caracterizando o problema de segmentação semântica, a arquitetura conhecida como U-Net (RONNEBERGER; FISCHER; BROX, 2015) foi utilizada.

4.3.1 Redes Neurais Artificiais

As redes neurais artificiais foram desenvolvidas inspiradas na estrutura hierárquica e interconectada do modelo biológico neural (GOODFELLOW; BENGIO; COURVILLE, 2016). Estas redes são representadas por um empilhamento de neurônios os quais juntos definem uma camada. Os neurônios estão totalmente interligados (*fully connected*) a neurônios de camadas adjacentes por meio de conexões que são controladas por pesos. Durante o treinamento destas redes, busca-se calibrar o valor destes pesos uma vez que são eles que expressam a relevância da informação computada para a predição final.

Os neurônios são as unidades básicas de uma rede neural e têm o papel de realizar operações lineares em que pesos (w) são multiplicados por um valor de entrada (x) e adicionados a um valor de *bias* (b) (NG et al., 2014) . O valor obtido em cada neurônio é multiplicado pela função de ativação não linear (α) que favorece o aumento do grau de liberdade da função computada. Consequentemente, permite o aprendizado de padrões complexos e não lineares expressos nos dados (equação

1 e figura 4). Os pesos são valores que variam durante o processo de treinamento da rede e vão sendo calibrados para definir os valores mais relevantes para a predição do modelo. Já o valor de *bias* funciona como uma constante que permitirá o deslocamento da função de ativação (KELLEHER, 2019).

$$y = \alpha(w \cdot x + b) \tag{1}$$
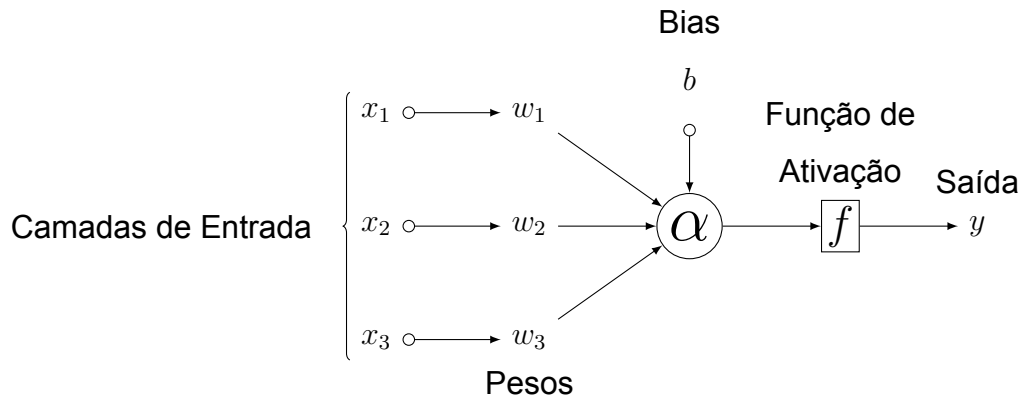
Figura 4: Operação realizada em cada neurônio de uma rede neural artificial.

Fonte: elaborado pelo autor.

O processo de transmissão de informação, ou *forward propagation*, em uma rede de várias camadas e neurônios, também conhecida como *multi-layer perceptron* (figura 5), ocorre por meio de cálculos de produtos escalares em cada camada da rede e passagem de informação para todos os neurônios da camada seguinte (KELLEHER, 2019). É importante ressaltar que uma vez que o *output* de uma camada anterior se torna o *input* da próxima camada, as computações vão se tornando sucessivamente mais complexas e abstratas à medida em que avançam até o último neurônio da rede.
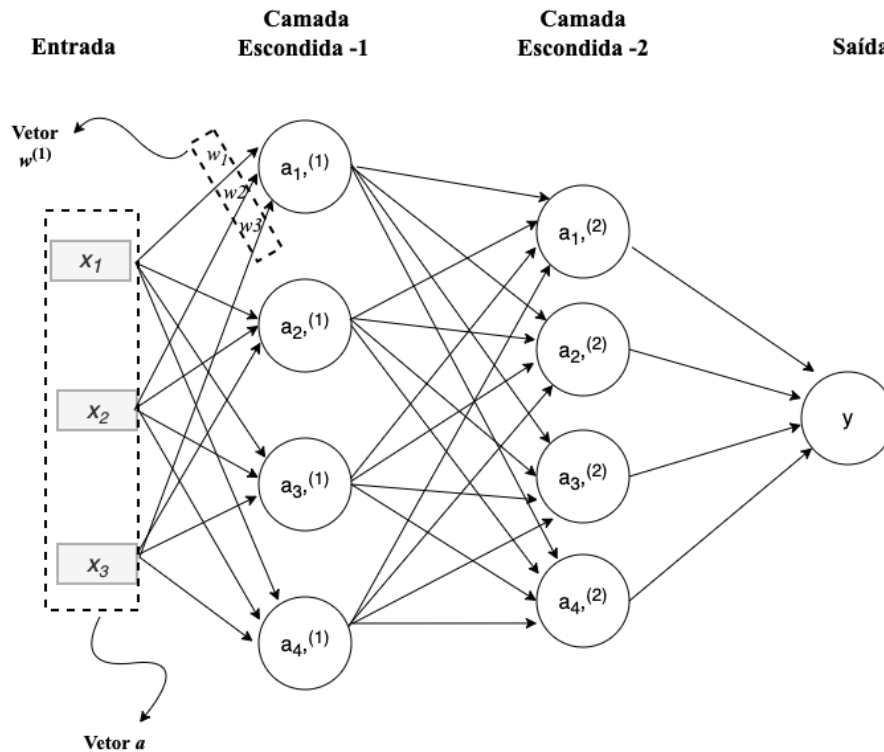
Figura 5: Rede neural de múltiplas camadas.

Fonte: elaborado pelo autor.

Para elucidar este processo, vemos que o neurônio $a_1{,}^{(1)}$ (figura 5) irá computar um produto escalar entre o vetor que contém os pesos (*w*), que estão ligados a estes neurônios, e o vetor (*a*), que contém os *inputs* (advindos dos *inputs* x1, x2, x3) e adicioná-los a um valor de *bias* (*b*) (eq. 2). A este resultado será aplicada uma função de ativação (α). Posteriormente, a informação calculada é transmitida para todos os neurônios da camada seguinte ($a_1{,}^{(2)}$) que realizará uma operação similar (eq. 3), porém, agora o *input* representa valores computados pelos neurônios da camada anterior ($a^{(1)}$). O cálculo realizado em cada camada pode ser generalizado conforme a equação 4. Onde "L" representa a camada em que o neurônio está localizado e "W" a matriz que contém os pesos ligados a cada neurônio, sendo cada linha desta matriz correspondente a um neurônio da rede.

$$a1{,}^{(\,1\,)} = \alpha(w^{(1)} \cdot a^{(0)} + b1) \tag{2}$$

$$a1{,}^{(\,2\,)} = \alpha(w^{(2)} \cdot a^{(1)} + b1) \tag{3}$$

$$a^{(L)} = \alpha(W^{(L)} \cdot a^{(L-1)} + b^{(L-1)}) \tag{4}$$

Em redes que utilizam o processo de treinamento supervisionado, como a descrita neste trabalho, o último neurônio é responsável por calcular um valor escalar que permitirá avaliar se o valor determinado pelo modelo é igual ou próximo ao valor esperado. Essa comparação entre o valor esperado e o previsto pelo modelo é realizada por uma função de custo. Com base nessa função, ocorrerá o processo de propagação reversa (*backward propagation*) que por meio do cálculo do gradiente dos pesos irá ajustá-los, em cada época de treinamento [3], visando reduzir o valor da função de custo.

A função de custo (*J*) é uma função diferenciável que é utilizada para mensurar a qualidade da predição que está sendo realizada (CHARU, 2018). A função de erro médio quadrático será utilizada aqui como exemplo para simplificar a explicação. Na sessão 4.3.3 a função de entropia cruzada binária utilizada neste trabalho é explicada. Na equação 5, "*w*" representa todos os pesos da rede, "*b*" os valores de *bias*, "*n*" o total de dados de treino e "*y*" o valor de saída da rede, "$\hat{Y}$" o valor esperado. Note que nessa função o valor nunca é negativo e se aproxima de zero à medida que o valor predito pelo modelo se aproxima do valor esperado.

$$J(w, b) = \frac{1}{2n} \sum_x (y - \hat{Y})^2 \tag{5}$$

Após o cálculo da função de custo, tem início a etapa de *backward propagation* em que os gradientes das funções computadas em cada neurônio são calculados de maneira reversa, isto é, da camada de saída da rede até a primeira camada escondida com base na regra da cadeia do Cálculo. A regra da cadeia é utilizada aqui, pois uma vez que as funções computadas nos neurônios são compostas de funções computadas em neurônios anteriores, a diferenciação deve seguir a lógica da equação 6, em que a derivada da função externa (*f(x)*) é multiplicada pela função interna (*g(x)*). O intuito deste processo é calcular o gradiente de cada neurônio e com isso mover os parâmetros (pesos) da rede na direção negativa do maior gradiente para reduzir o valor da função de custo.

---

[3]Uma época de treinamento é definida quando todos os dados disponíveis são utilizados no treinamento do modelo.

$$\frac{d}{d_w} f(g(w)) = f'(g(w)).g'(w) \tag{6}$$

A figura 6 mostra uma rede neural simplificada com apenas uma camada escondida que auxiliará a ilustrar todo o processo de treinamento. Na etapa de *forward propagation* o neurônio $a_1,^{(1)}$ irá computar a soma ponderada dos pesos (*w*) pelos valores de entrada (*x*), expressa na equação 7. O neurônio $a_2,^{(1)}$ realizará o mesmo processo conforme descrito na equação 8. Ao final, o resultado passará por uma função de ativação (α). O neurônio de saída (*y*) irá receber o resultado dos neurônios anteriores ($a_1,^{(1)}$ e $a_2,^{(1)}$) e realizará o cálculo conforme descrito na equação 9. Ao final, a função de custo será calculada utilizando o valor escalar obtido na saída da rede e o valor esperado ($\hat{Y}$).
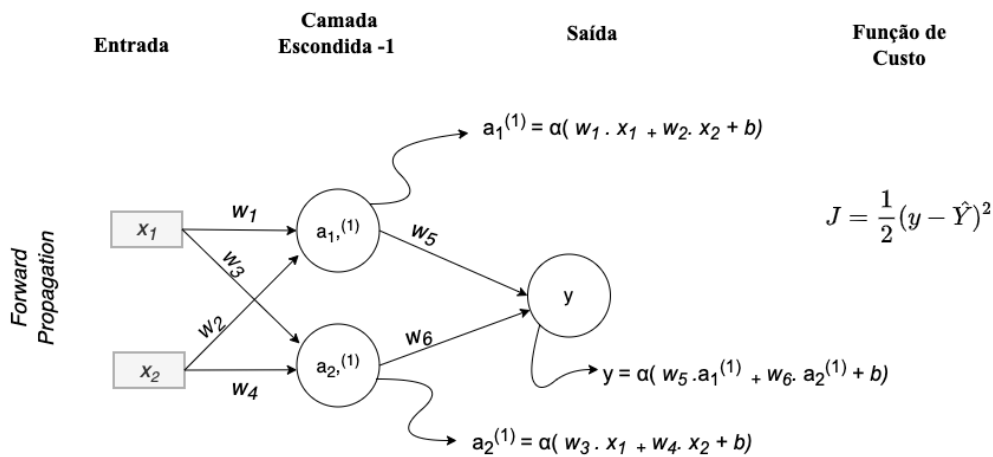
Figura 6: Etapa de *forward progation*.

Fonte: elaborado pelo autor.

$$a_1,^{(1)} = \alpha(w_1.x1 + w_2.x_2 + b_1) \tag{7}$$

$$a_2,^{(1)} = \alpha(w_3.x_1 + w_4.x_2 + b_1) \tag{8}$$

$$y = \alpha(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2) \tag{9}$$

Após o cálculo do valor da função de custo, tem início a etapa de *backpropagation* que será explicada em detalhes, pois trata-se da parte mais complexa e

crucial para o funcionamento de uma rede neural. Assim, a função de custo (J) é derivada com relação aos pesos (w), seguindo a regra da cadeia (eq. 10):

$$\frac{\partial J}{\partial w} = (y - \hat{Y})^{2\prime} = \frac{2}{2}(y - \hat{Y}).y' - \hat{Y} \tag{10}$$

A derivada da função y (eq. 9) seguirá as mesmas regras já que também se trata de uma função composta. O cálculo ocorre de acordo com a equação 11. Note que $\hat{Y}$ é uma constante, portanto, deixa de existir nessa função já que sua derivada é zero.

$$\frac{\partial J}{\partial w} = (y - \hat{Y}).y' = (y - \hat{Y})\alpha(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2)' \tag{11}$$

Para prosseguir com a derivação é necessário definir a função de ativação (α). Assim, visando facilitar o entendimento, a função sigmóide (eq. 12) será considerada em todos neurônios [4].

$$\alpha(w^T x + b) = \frac{1}{1 + e^{-(w^T x + b)}} \tag{12}$$

Prosseguindo com os cálculos (eq. 13) e derivando a função em relação a w_5, os valores de w_6 e do *bias* (b) são considerados constantes, portanto, seu valor é 0 (zero). Já o valor da derivada de w_5 é 1 (um), pois, conforme a regra da identidade, a derivação se dá em relação a este valor (eq. 14)

$$\frac{\partial J}{\partial w} = (y - \hat{Y})\alpha'(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2).(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2)' \tag{13}$$

$$\frac{\partial J}{\partial w} = (y - \hat{Y})\alpha'(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2).a_1^{(1)} \tag{14}$$

Sabendo que a derivada da função sigmóide é definida conforme a equação 15 e substituindo "z" por "$w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2$", se obtém a equação 16:

$$\alpha' = \alpha(z).(1 - \alpha(z)) \tag{15}$$

$$\frac{\partial \alpha}{\partial w} = \alpha(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2).(1 - \alpha(w_5.a_1,^{(1)} + w_6.a_2,^{(1)} + b_2)) \tag{16}$$

---

[4]Neste trabalho, a fução sigmóide foi utilizada apenas no último neurônio da rede. Nos demais neurônios a função *Rectified Linear Unit (ReLU)* foi utilizada. Ambas as funções estão explicadas em detalhes na seção 4.3.2.

Note que as equações desta função correspondem justamente ao valor computado na camada de saída da rede (y), assim, pode-se substituir este valor e obter a equação 17. Juntando o resultado desta derivação com o que foi obtido anteriormente, chega-se na equação 18.

$$\frac{\partial \alpha}{\partial w} = y.(1-y) \tag{17}$$

$$\frac{\partial J}{\partial w} = (y - \hat{Y})y.(1-y).a_1^{(1)} \tag{18}$$

Este resultado define o gradiente do peso w_5 (eq. 19) e analogamente o gradiente de w_6 (eq. 20). Note que apenas o valor $a_2^{(1)}$ é diferente, em função da derivação da equação (14) em relação a w_6 e não mais a w_5. O pensamento análogo pode ser realizado para neurônios que se encontram em camadas escondidas anteriores (Fig. 7).

$$\nabla w_5 = (y - \hat{Y}).y.(1-y).a_1^{(1)} \tag{19}$$

$$\nabla w_6 = (y - \hat{Y}).y.(1-y).a_2^{(1)} \tag{20}$$

$$\nabla w_3 = (y - \hat{Y}).y.(1-y).w_5.a_1^{(1)}(1-a_1^{(1)})x_2$$
$$\nabla w_1 = (y - \hat{Y}).y.(1-y).w_5.a_1^{(1)}(1-a_1^{(1)})x_1$$
$$\nabla w_5 = (y - \hat{Y}).y.(1-y).a_1^{(1)}$$
$$\frac{\partial J}{\partial w} = (y - \hat{Y}).\sigma'(w_5.a_1^{,(1)} + w_6.a_2^{,(1)} + b_2).a_1^{(1)}$$
$$\nabla w_6 = (y - \hat{Y}).y.(1-y).a_2^{(1)}$$
$$\nabla w_4 = (y - \hat{Y}).y.(1-y).w_6.a_2^{(1)}(1-a_2^{(1)})x_2$$
$$\nabla w_3 = (y - \hat{Y}).y.(1-y).w_6.a_2^{(1)}(1-a_2^{(1)})x_1$$
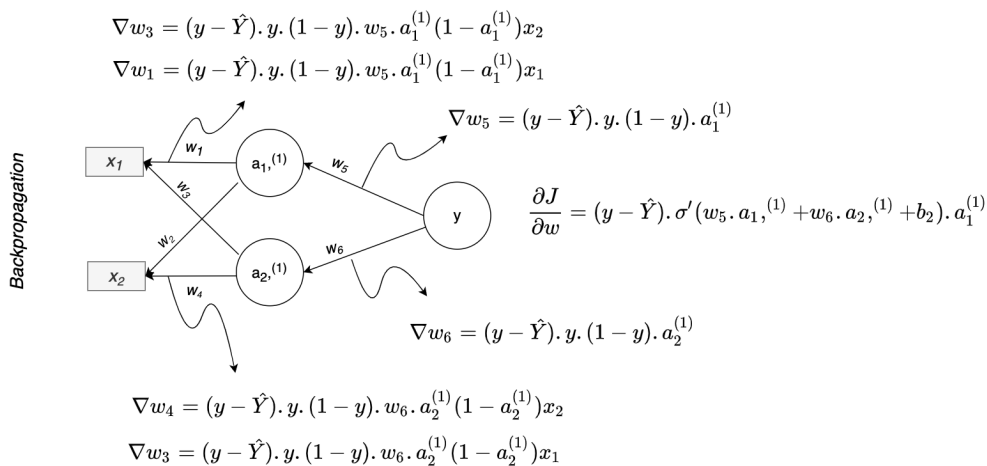
Figura 7: Etapa de *backpropagation*.

Fonte: elaborado pelo autor.

Note que o cálculo do gradiente permite determinar qual a direção que irá direcionar a função de custo a ter seu valor reduzido, porém é importante notar que a

magnitude deste ajuste não pode ser definida pelo processo de derivação, assim o parâmetro conhecido como *learning rate* definirá esta magnitude (figura 8).
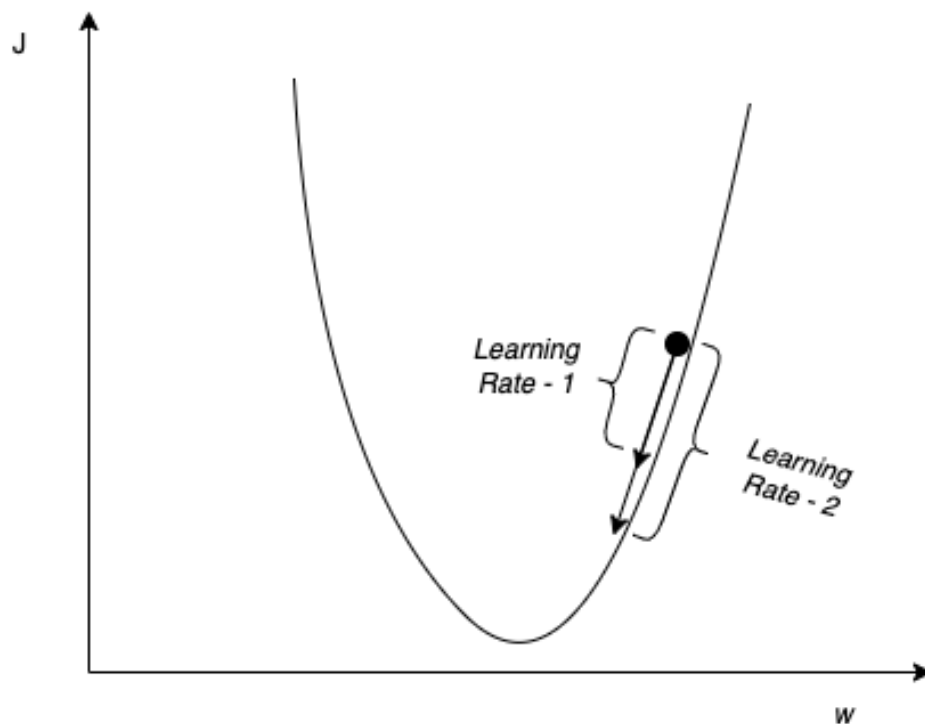
Figura 8: Impacto do valor de *learning rate* no processo de otimização.

Fonte: elaborado pelo autor.

O learning rate (α) é um valor estabelecido durante o processo de treinamento que determina qual a magnitude do ajuste que o algoritmo de otimização *gradient descent* [5] realizará nos pesos de cada neurônio (eq. 21).

$$w = w - \alpha \cdot \frac{\partial J}{\partial w} \tag{21}$$

Com os ajustes nos pesos realizados, o processo de treinamento se repete por um número pré-determinado de épocas ou até que uma determinada condição seja cumprida. A definição deste valor e a calibração de outros parâmetros (*hyper-parameters*) estão diretamente ligados a performance do modelo. Tais parâmetros são definidos manualmente e buscam evitar que o modelo apresente *overfitting* (ver

---

[5]Para facilitar a explicação, o otimizador conhecido como *gradient descent* será utilizado neste exemplo, porém, existem otimizadores mais avançados como o *Adaptive Moment Estimation* (Adam) que foi utilizado na pesquisa e está explicado em detalhes na seção 4.3.5.

seção 4.3.6), isto é, permitir que o modelo realize predições corretas em dados que não foram utilizados em seu treinamento.

4.3.2 Funções de Ativação

As funções de ativação são funções diferenciáveis que transformam o sinal de *input* em *output*, adicionando um componente não linear a esta transformação. Tal resultado guia o processo de ativação ou não ativação de um neurônio. Na etapa de *backpropagation*, a diferenciação destas funções auxiliam o processo de optimização dos pesos. As funções sigmóide, tangente hiperbólica (TanH), unidade linear retificada (ReLU), *Leaky* ReLU e unidade linear exponencial (ELU), geralmente são as mais utilizadas como funções de ativação. Neste trabalho foram utilizadas as funções ReLU e sigmóide.

A função sigmóide (figura 9), ou logística, transforma seu *input*, que se encontra no domínio dos números reais, em valores que estão no intervalo (0,1). Assim, geralmente ela é utilizada no neurônio final em problemas de classificação binária. A derivada apresenta o valor máximo (0.25) quando o valor de *input* é zero e à medida que o valor diverge de zero em ambas as direções, a derivada se aproxima de zero.
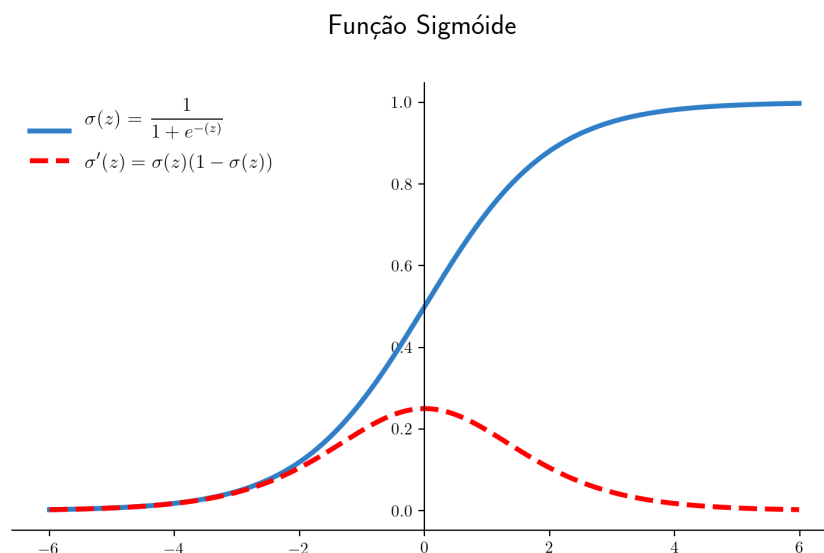
Figura 9: Gráfico da função sigmóide e de sua derivada de primeira ordem.

Fonte: elaborado pelo autor.

A função ReLU (figura 10) mantém apenas os elementos positivos do *input* e transforma os valores negativos em zero. Devido a simplicidade desta transforma-

ção não linear, e boa performance em problemas de predição, esta função é comumente utilizada nos neurônios localizados nas camadas escondidas (QUINN et al., 2019). Esta função apresenta derivada de valor zero quando o *input* é negativo e um (1) quando o valor é positivo.
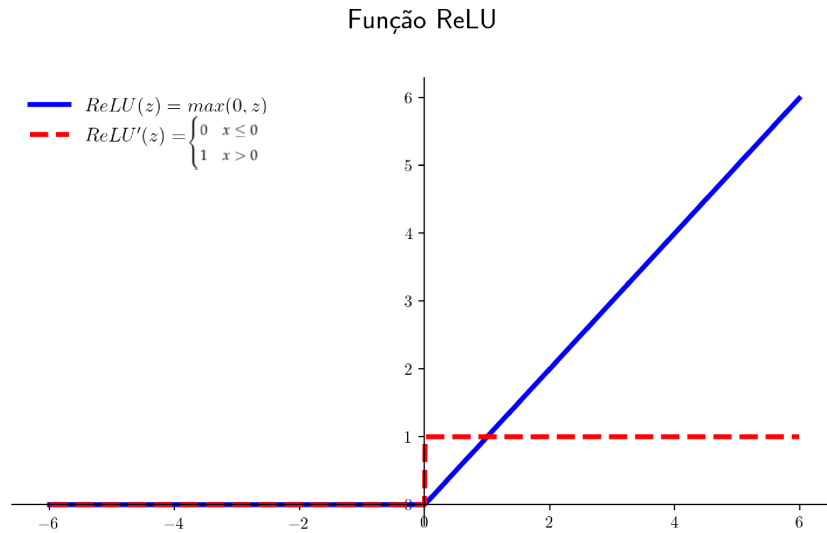
Figura 10: Gráfico da função ReLU e de sua derivada de primeira ordem.

Fonte: elaborado pelo autor.

4.3.3 Função de Custo

A função de custo (J) é utilizada para mensurar a qualidade da predição que está sendo realizada. O treinamento de modelos supervisionados tem como objetivo minimizar o valor desta função. Em problemas de segmentação binária é comum o uso da função de entropia cruzada binária (eq. 22). Nesta equação, Y representa o valor esperado, $\hat{Y}$ o valor predito pelo modelo, N o número de amostras.

$$J = -\frac{1}{N}\sum_{i=1}^{N}(Y)(-log(\hat{Y}) + (1 - Y)(-log(1 - \hat{Y})) \tag{22}$$

4.3.4 *Learning Rate*

A atualização dos pesos entre os neurônios é feita pelo algoritmo de otimização, visando encontrar um valor mínimo, que pode ser local ou global, da função de custo. A dimensão do ajuste é controlada pelo *learning rate*, por isso, é importante encontrar um valor ideal de *learning rate*, já que valores muito elevados podem dificultar a convergência, enquanto valores baixos podem levar muito tempo

para convergir (figura 11).

Muito baixo	Correto	Muito Elevado
Quando o valor do *Learning Rate* é muito baixo o processo de op-timização pode levar muito tempo até clegar em um ponto mínimo.	Um valor adequado de *Learning Rate* permite que o modelo chegue em um ponto mínimo rapidamente.	Um valor muito elevado de *Learning Rate* gera mudanças drásticas nos valores dos pesos, o que dificulta a convergência do modelo.

Figura 11: Impacto de diferentes valores de *learning rate* na convergência do modelo.

Fonte: Modificado de https://www.jeremyjordan.me/nn-learning-rate/, acessado em 20/06/2020.

4.3.5 Função de Otimização

A função de otimização tem como objetivo ajustar os pesos da rede neural, visando reduzir a função de custo. A função *Adaptative Movement Estimation* (Adam) (KINGMA; BA, 2014) foi utilizada nesta pesquisa. Este algoritmo foi desenvolvido para acelerar o processo de otimização e é considerado o melhor para diversos problemas em que se utiliza a técnica de *deep learning* (BUDUMA; LOCASCIO, 2017).

A função Adam é um algoritmo que utiliza características dos algoritmos *Adaptive Gradient Algorithm* (AdaGrad) e *Root Mean Square Propagation* (RMSProp) e com isso consegue realizar um ajuste para o valor de *learning rate* dinamicamente para cada parâmetro da rede. Além disso, guarda o decaimento exponencial do quadrado dos gradientes anteriores. Tais características facilitam a convergência dos modelos, já que um valor de learning rate fixo é desvantajoso quando a função de custo se aproxima de valores mínimos e guardar o valor dos decaimento dos gradientes melhora a eficiência computacional e a efetividade do algoritmo (GOOD-FELLOW; BENGIO; COURVILLE, 2016). Para uma explicação matemática mais detalhada recomenda-se a leitura de Kingma e Ba (2014).

4.3.6 *Overfitting* e *Underfitting*

Ao treinar um modelo é importante que este apresente boa capacidade de generalização, isto é, classifique corretamente dados que não participaram do processo de treinamento. Para isto, comumente o *dataset* é separado em dados de treino, validação e teste. Estes dados são utilizados para avaliar se está ocorrendo o processo de *underfitting* ou *overfitting* (figura 12).
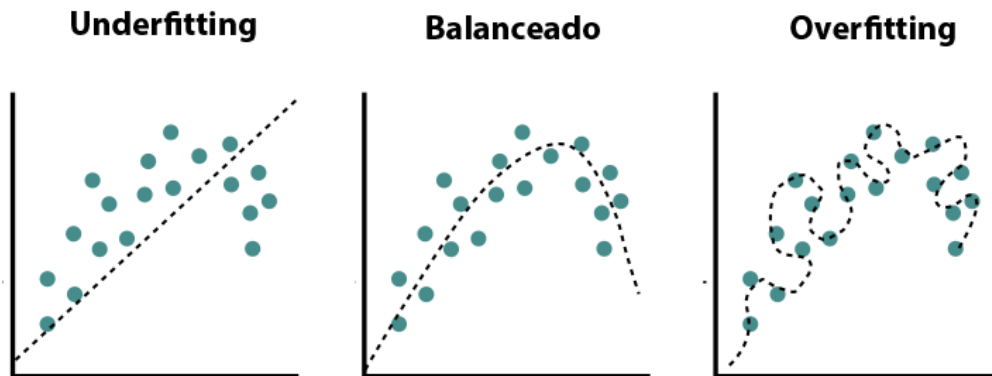
Figura 12: Processo de *overfitting* e *underfitting*.

Fonte: Modificado de https://www.educative.io/edpresso/overfitting-and-underfitting, acessado em 20/06/2020.

O *underfitting*, ou alto *bias*, ocorre quando o modelo obtido comete muitos erros ao ser avaliado nos dados em que foi treinado. Isto mostra que o modelo utilizado é muito simples ou ainda que os dados apresentados são insuficientes para o modelo aprender representações relevantes ao problema. A utilização de redes mais complexas – por exemplo, uma rede neural com mais camadas escondidas – pode ser uma alternativa para resolver este problema.

Por outro lodo, no *overfitting*, ou alta variância, o modelo comete poucos ou nenhum erro nos dados de treino, porém ao ser avaliado em um dado de teste o erro é alto. Isto pode evidenciar que o modelo é muito complexo, permitindo o ajuste perfeito aos dados de treino, ou que a quantidade de dados de treinamento não é suficiente. Para reduzir o overfitting, é comum o uso de camadas de regularização (*dropout* e *batch normalization*), *data augmentation* e transferência de conhecimento (*transfer learning*) (BURKOV, 2019).

4.4 Redes Neurais Convolucionais

Redes Neurais Convolucionais (RNC) (*Convolution Neural Networks - CNN*) foram desenvolvidas inspiradas no modo de funcionamento do sistema de visão animal. Conforme descrito pelos trabalhos de Hubel e Wiesel (1959) e Marr (1982), o córtex visual possui uma organização complexa de células que atuam como filtros em diferentes sub-regiões do campo visual. Este processo de filtragem ocorre de modo hierárquico em que as informações mais complexas de um objeto como profundidade e volume são construídas a partir de características mais simples tais como bordas, curvas e vértices. Estes primeiros modelos serviram de base para o desenvolvimento das RNC.

As redes convolucionais são consideradas como um tipo de rede neural artificial de múltiplas camadas (LECUN et al., 1989). Assim, seu processo de treinamento é análogo ao das redes neurais artificiais. Porém, os neurônios nas redes convolucionais são representados por pequenos filtros e não estão totalmente conectados uns aos outros. Tais características permitem que estas redes mantenham a estrutura espacial dos dados de entrada, não sendo necessário transformá-los em um vetor. Além disso, as representações apreendidas são locais, ao invés de representações globais como nas redes densas (CHOLLET, 2017). Com representações locais o modelo pode detectar um mesmo padrão em diferentes posições da imagem. Ainda, as representações obtidas apresentam uma hierarquia de complexidade, em que as camadas iniciais da rede se especializam em detectar feições simples, a saber, bordas verticais, bordas horizontais e vértices, e as camadas mais profundas combinam estas informações para obter padrões sucessivamente mais complexo e abstratos.

A arquitetura de uma rede de convolução é composta por uma intercalação de camadas de convolução e *pooling*. As camadas de convolução são pequenas matrizes de números reais, em geral de dimensões ímpares (3x3, 5x5, 7x7), que funcionam como filtros, também conhecidos como *kernels*. Estes filtros se deslocam por todas as dimensões da imagem realizando produtos escalares que geram um mapa de características (*feature map*) da imagem em análise. O treinamento deste tipo de rede visa calibrar os valores destes pesos para que os filtros obtenham características relevantes que permitam identificar determinado elemento na imagem. Em

contrapartida, as camadas de *pooling* são utilizadas para reduzir o custo computacional já que diminuem a dimensão espacial do mapa de característica obtido pelas operações de convolução e auxiliam a remover valores redundantes e preservar os mais relevantes. Para isso, pequenas janelas móveis, geralmente de dimensões pares (2x2, 4x4, 6x6), deslocam-se por toda a imagem extraindo o valor máximo dos elementos englobados por esta.

4.4.1 Camadas de Convolução

As camadas de convolução atuam como pequenos filtros que se deslocam por toda a imagem realizando produtos escalares (figura 13). O resultado desta operação passa por uma função de ativação não linear que gera um mapa de característica que contém representações abstratas dos dados de entrada. Os mapas gerados pelos diversos filtros presentes em uma camada são empilhados formando um tensor cuja profundidade é igual ao número de filtros.
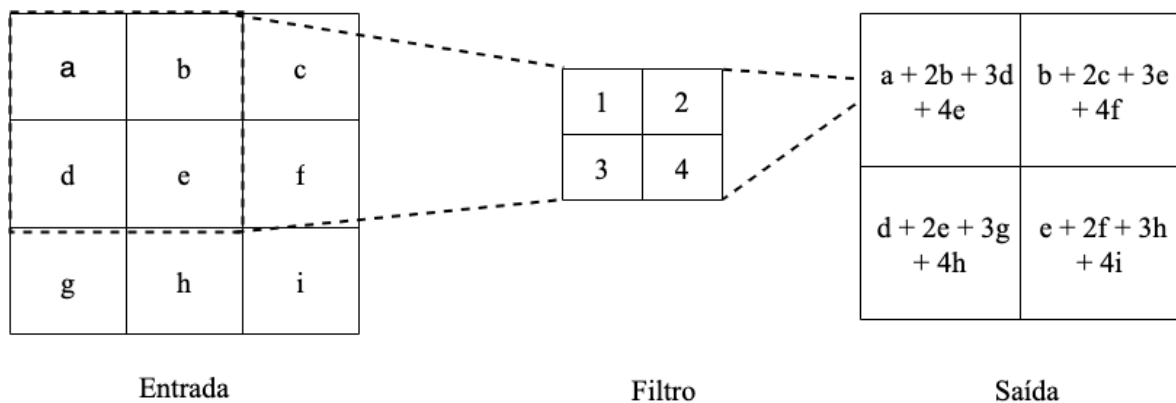
Figura 13: Operação de convolução.

Fonte: elaborado pelo autor.

As camadas de convolução têm como *input* uma matriz tridimensional que possui largura, altura e profundidade. A largura e altura representam as dimensões espaciais da imagem, já a profundidade caracteriza a quantidade de canais de cor[6] presentes na imagem. Os filtros de convolução que irão operar nesta imagem, também apresentarão três dimensões, sendo que a largura e a altura representam as dimensões espaciais do filtro e a profundidade será a mesma da imagem de *input*, permitindo que este filtro opere ao longo de todo o volume da imagem (figura 14).

---

[6]Em sensoriamento remoto a profundidade é equivalente a quantidade de bandas espectrais presentes no sensor.
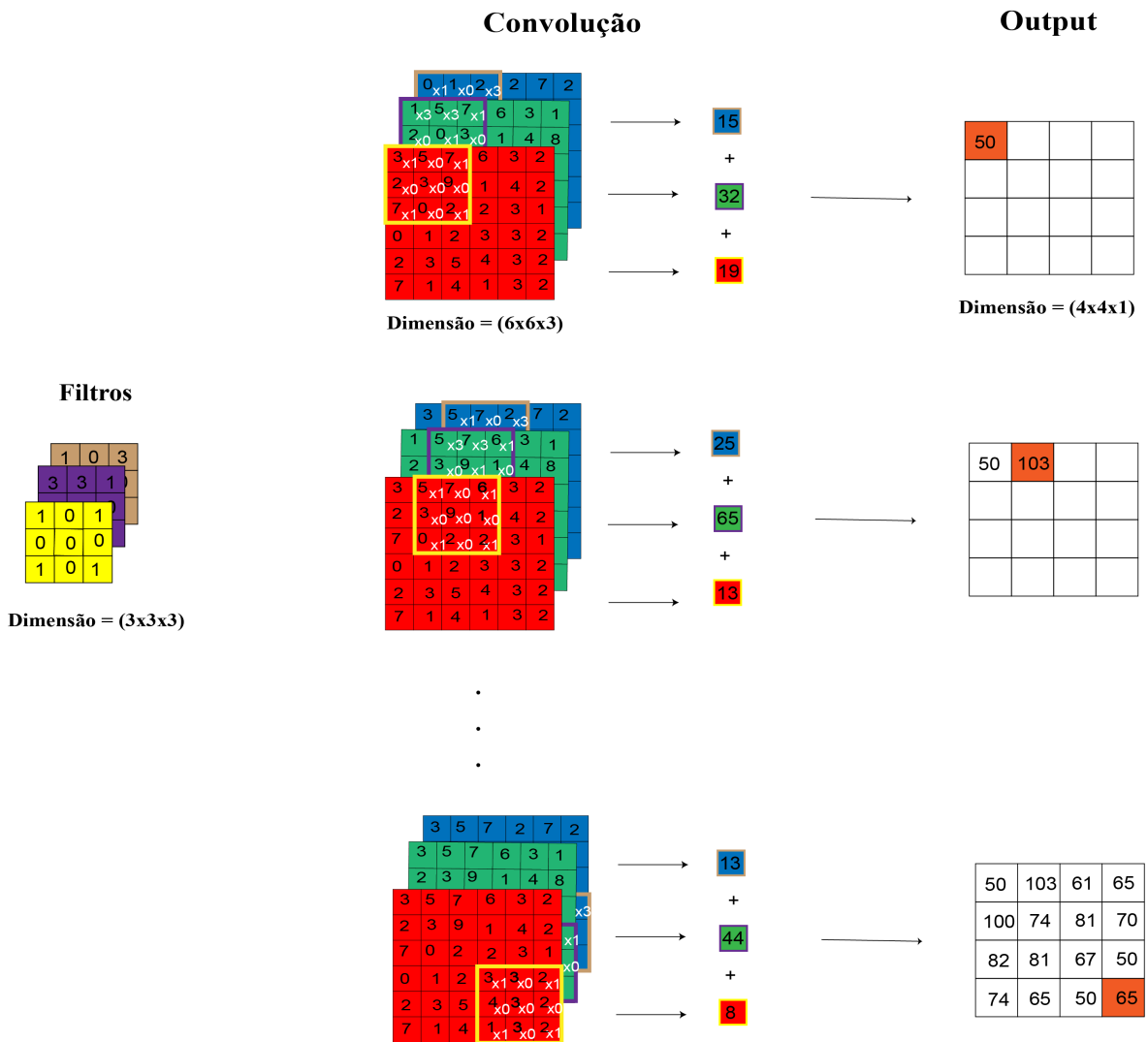
Figura 14: Operação de convolução em uma imagem RGB.

Fonte: elaborado pelo autor.

O filtro irá se movimentar ao longo da imagem com base no *stride* que é um parâmetro que define qual o deslocamento lateral, em pixels. O controle deste parâmetro em conjunto com o *padding*, que representa a adição de pixels nas bordas da imagem, define qual será a dimensão de largura e altura resultante de cada camada de convolução (eq. 23).

$$O = \frac{(w - K + 2P)}{S} + 1 \qquad (23)$$

Nesta equação, "O" representa a dimensão de saída da operação de convolução, "W" a dimensão (altura ou largura) do dado de entrada, "K" a dimensão do filtro, "P" o valor de *padding* e "S" o valor do *stride*.

4.4.2 Camadas de *Pooling*

As camadas de *pooling* reduzem a dimensão espacial dos mapas obtidos nas camadas de convolução. Essa operação tem como objetivo reduzir o custo computacional e, além disso, realçar os valores que mais ativaram os *kernels*. Conceitualmente, esta operação é similar a de convolução, porém, os filtros, ao invés de realizarem transformações lineares, extraem o maior valor (*max pooling*) ou o valor médio (*average pooling*) dos elementos englobados pela janela de filtragem. Além disso, comumente as janelas de *pooling* apresentam dimensão 2x2 e *stride* 2 para reduzir o mapa de características pela metade.
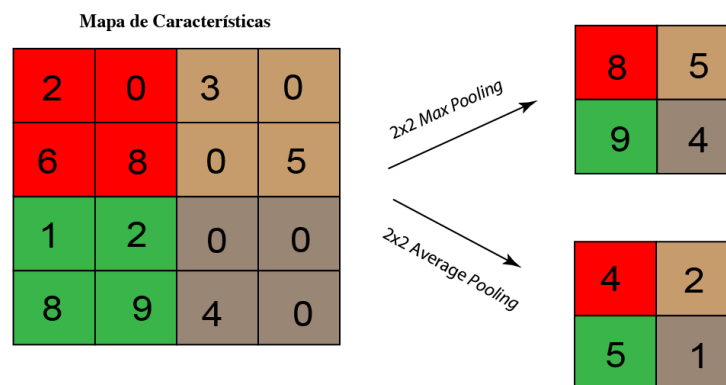
Figura 15: Operações de *Max Pooling* e *Average Pooling*

Fonte: elaborado pelo autor.

4.4.3 Camadas de *Dropout*

O *dropout* (SRIVASTAVA et al., 2014) é uma técnica utilizada durante a fase de treinamento da rede que envolve a desativação aleatória com probabilidade p de uma parcela de neurônios do modelo. A ideia desta técnica é reduzir a complexidade do modelo, consequentemente, o *overfitting*.

4.4.4 Camadas de *Batch Normalization*

O *batch normalization* (IOFFE; SZEGEDY, 2015) é uma técnica utilizada para estabilizar a distribuição dos dados, em um grupo (*batch*) de amostras, antes do *input* destes valores na próxima camada. Para isso, calcula-se a média acumulada dos *outputs* de uma determinada camada e este valor é dividido pelo desvio padrão do grupo. Segundo Santurkar et al. (2018) este procedimento suaviza o "terreno" da função de optimização, o que permite o uso de maiores valores de *learning rate* e uma convergência mais rápida do modelo.

4.5 Segmentação Semântica

A segmentação semântica é definida como um problema de classificação em que se busca classificar, em classes pré-definidas, os pixels de uma imagem. Em modelos supervisionados, deve-se realizar uma classificação prévia dos pixels das imagens de treino, que posteriormente servirá como base para a geração de uma máscara que indicará as classes correspondentes de cada pixel. Esta imagem deve apresentar as dimensões de largura e altura da imagem de treino. Uma vez que se busca classificar cada pixel da imagem, o resultado deve apresentar a mesma dimensão espacial das camadas de *input*. Assim, as redes utilizadas para segmentação geralmente utilizam apenas camadas convolucionais em sua estrutura, sendo conhecidas como redes totalmente convolucionais (*fully convolutional networks*).

As redes totalmente convolucionais comumente apresentam uma estrutura que contém um caminho contracional (*encoder*), o qual é responsável por capturar o contexto da imagem, porém não leva em conta a localização espacial do pixel classificado. Para recuperar a posição espacial são realizadas operações de *upsampling* e concatenação em uma rede de expansão (*decoder*). Em arquiteturas como a U-net existem conexões que ligam as representações obtidas durante o caminho contracional ao caminho extensional por meio de concatenações.

4.5.1 U-Net

A U-net (RONNEBERGER; FISCHER; BROX, 2015) (figura 16) é uma rede de segmentação totalmente convolucional desenvolvida com intuito de ser treinada de ponta a ponta (*end-to-end*) com poucas imagens. Esta rede classifica cada pixel da imagem por meio de uma arquitetura do tipo *encoder-decoder*. A posição espacial da classificação é recuperada na etapa *decoder* por um processo de *upsampling*, através de convoluções transpostas e concatenação dos mapas de características, localizados em posições simétricas no caminho contracional. É devido a esta simetria, similar a letra "U", que a rede possui o nome de U-net.
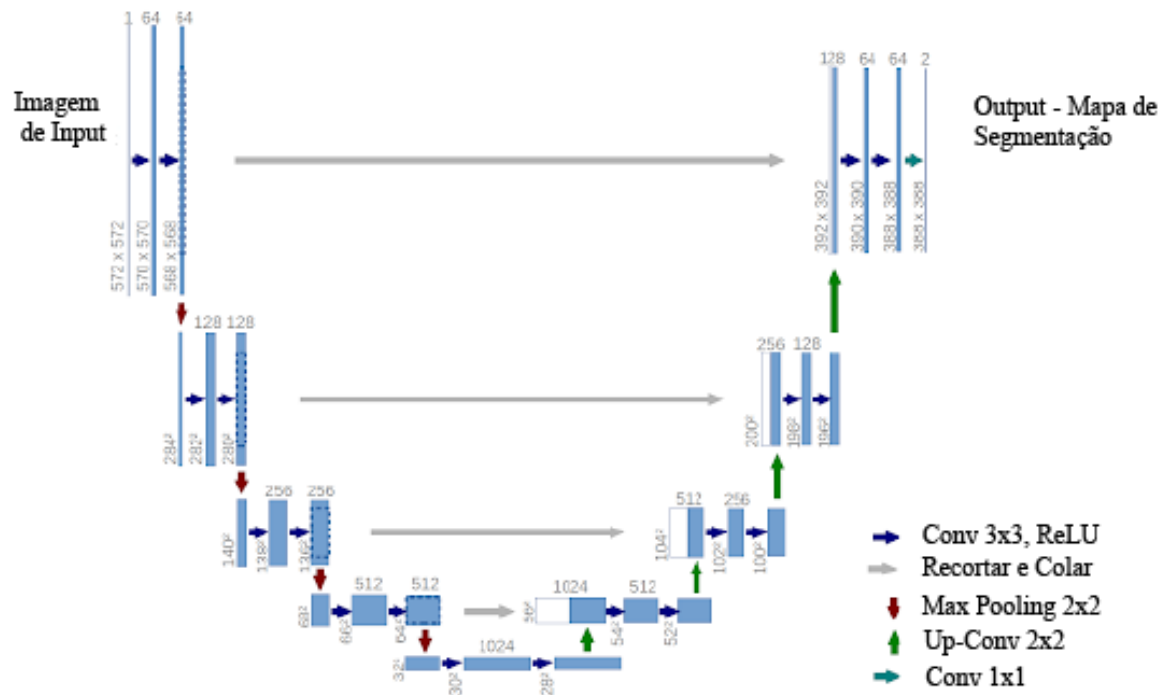
Figura 16: Arquitetura de rede U-net.

Fonte: Modificado de Ronneberger, Fischer e Brox (2015).

4.6 Operações Morfológicas

As operações morfológicas foram utilizadas neste trabalho para melhorar os resultados obtidos pelos modelos de *deep learning*. Uma vez que explicações detalhadas sobre a teoria matemática morfológica fogem ao escopo deste trabalho, recomenda-se a leitura de Gonzalez e Woods (2008) para explicações com maior rigor e detalhe matemático. Nesta seção, o foco principal será explicar o efeito de cada uma das operações morfológicas nas máscaras binárias que resultam dos modelos de *deep learning*.

A morfologia matemática surgiu em função de pesquisas conjuntas de G. Matheron e J. Serra que utilizaram a Teoria dos Conjuntos para definir suas operações. Com isso, a descrição completa de uma imagem binária é caracterizada pelo conjunto de todos os seus pixels pretos (MATHERON; SERRA, 2002). As operações morfológicas consistem em percorrer a imagem com um elemento estruturante e quantificar a forma que este se encaixa na imagem. Assim, podem ser realizadas operações de erosão, dilatação, abertura e fechamento (fig. 17). Estas operações são tipicamente aplicadas na remoção de imperfeições introduzidas durante a segmentação.

Imagem Original

Erosão   **Dilatação**

Abertura   **Fechamento**
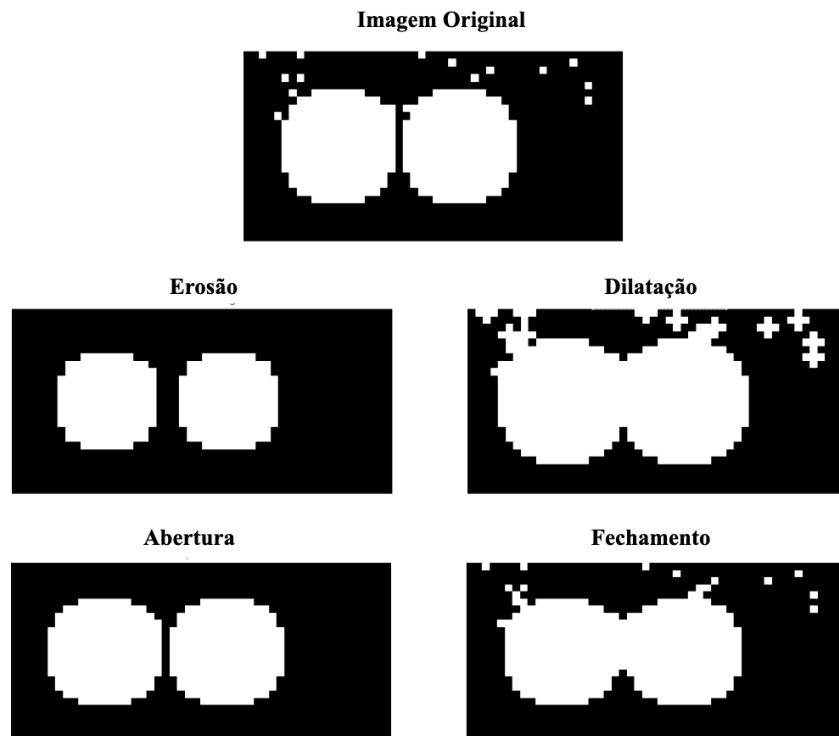
Figura 17: Operações morfológicas

Fonte: elaborado pelo autor.

O elemento estruturante é uma matriz que possui apenas valores "1"e "0"e que possui dimensões menores que a imagem sob análise. Seu pixel central, ou origem, percorre cada elemento da imagem identificando o pixel que será processado e sua vizinhança. O efeito do processamento é similar a um processo de filtragem e depende da forma do operador que está sendo usado (fig. 18).
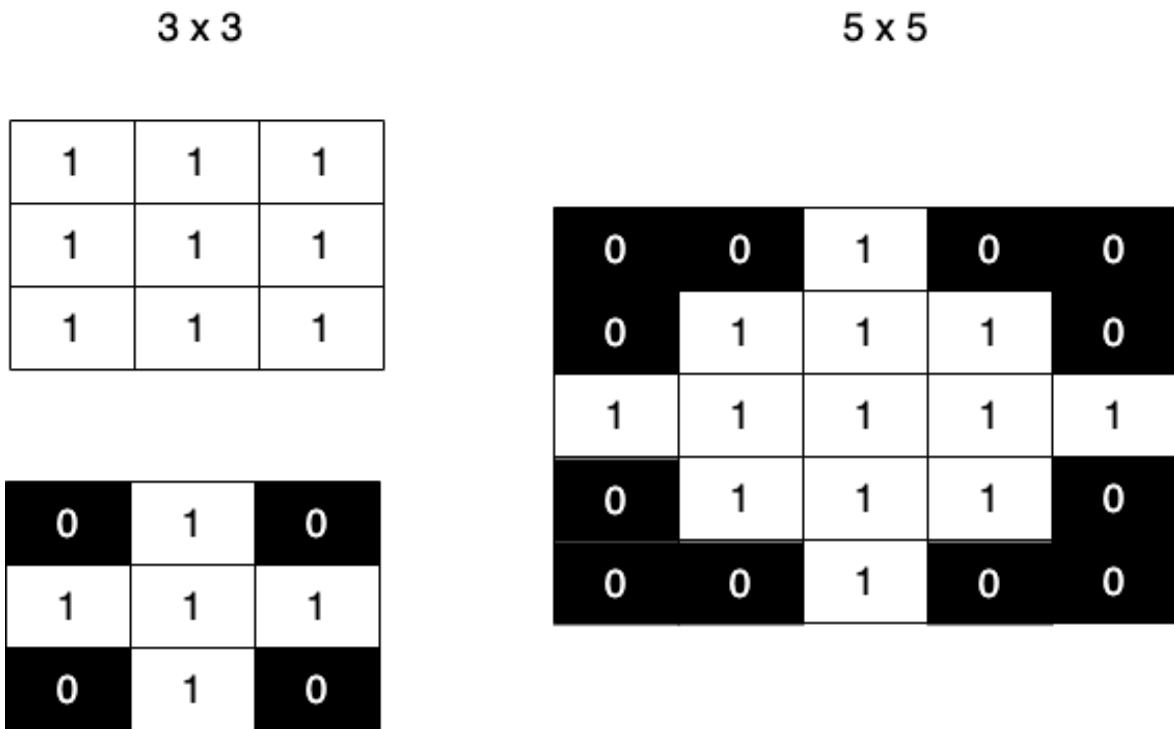
Figura 18: Elementos estruturantes com diferentes dimensões e formas.

Fonte: elaborado pelo autor.

4.6.1 Erosão e Dilatação

A operação de erosão é utilizada para escurecer a imagem por meio da ampliação do plano de fundo (pixels pretos) o que auxilia a remover componentes menores que o elemento estruturante, além disso, ajuda a separar componentes conectados. Pode-se definir a erosão pela equação 24, em que $A \ominus B$ representa a erosão de A por B e "*z*" todos os pontos em que B, trasladados por *z*, contém A.

$$A \ominus B = \{z|(B)_z \subseteq A\} \tag{24}$$

A dilatação auxilia a clarear a imagem, remover os ruídos das máscaras e conectar pequenos buracos (pixels pretos) que se localizam em meio aos pixels brancos. Esta operação pode ser definida conforme a equação 25, em que $A \oplus B$ representa a operação de dilatação de A por B e "*z*" representa todos os deslocamentos em que $\hat{B}$ (reflexão de B sobre sua origem) e A se sobrepõem em ao menos um elemento (GONZALEZ; WOODS, 2008).

$$A \oplus B = \{z|[(\hat{B})_z \cap A] \subseteq A\} \tag{25}$$

4.6.2 Abertura e Fechamento

As operações de abertura geralmente auxiliam a suavizar os contornos de um objeto, quebrar pequenas ligações entre os componentes e eliminar o ruídos das imagens. Já as operações de fechamento também auxiliam a suavizar os contornos dos objetos, porém, ao contrário das operações de abertura, elas auxiliam a juntar às componentes com pequeno espaçamento, eliminar buracos em objetos e preencher vazios em seus contornos (GONZALEZ; WOODS, 2008).

A operação de abertura pode ser definida conforme a equação 26. Assim, a abertura de A por B é a erosão de A por B seguido por uma dilatação. Similarmente, o fechamento de A pelo elemento estruturante B é definido como a dilatação de A por B seguido pela erosão do resultado por B (eq. 27).

$$A \bullet B = (A \oplus B) \ominus B \tag{26}$$

$$A \circ B = (A \ominus B) \oplus B \tag{27}$$

5 RESULTADOS

A metodologia e os resultados obtidos neste trabalho foram organizados em dois artigos científicos. O artigo intitulado *"Landslide Segmentation with U-Net: Evaluating Different Sampling Methods and Patch Sizes"* foi elaborado durante o primeiro ano de pesquisa e disponibilizado no ArXiv na forma de *preprint* . Já o artigo elaborado durante o segundo ano de pesquisa e intitulado *"Landslide Segmentation with Deep Learning: Evaluating Model Generalization in Rainfall-Induced Landslides in Brazil"* foi submetido na revista indexada *Natural Hazards*. A seção abaixo apresenta um breve resumo destes dois trabalhos. Os textos completos podem ser consultados nos apêndices 7.1 e 7.2, respectivamente. Outros dois trabalhos também foram realizados em coautoria com pesquisadores da Universidade de Salzburg,na Áustria, e da Universidade de Wuhan, na China. Estes trabalhos podem ser consultados no apêndice 7.3 e 7.4.

5.1 *Landslide Segmentation with U-Net: Evaluating Different Sampling Methods and Patch Sizes*

O objetivo desta pesquisa foi avaliar como os diferentes métodos de amostragem e as dimensões das imagens de treinamento impactam a acurácia dos modelos de *deep learning* durante a segmentação de escorregamentos de terra. As imagens utilizadas para o treinamento e teste dos modelos são da região de Nova Friburgo, no Rio de Janeiro (RJ) e foram obtidas do sensor *RapidEye*. Foram também utilizados modelos digitais de elevação fornecidos pela *Alaska Satellite Facility* com 15 metros de resolução.. As imagens foram amostradas em dimensões com 32x32, 64x64 e 128x128 pixels utilizando *grids* regulares com sobreposição de 30% entre as grades e de maneira randômica. O processo de *data augmentation* também foi utilizado para aumentar a quantidade de amostras de treinamento artificialmente. Os modelos foram treinados utilizando a rede totalmente convolucional conhecida como U-Net. Os resultados mostram que os algoritmos que utilizaram imagens com maiores dimensões obtiveram maiores valores de precisão, *f1-score* e índice de Jaccard, enquanto os modelos treinados com menores dimensões obtiveram maiores valores de *recall*. Além disso, os modelos que utilizaram amostragem randômica obtiveram maiores acurácias nas áreas de teste avaliadas.

5.2 *Landslide Segmentation with Deep Learning: Evaluating Model Generalization in Rainfall-Induced Landslides in Brazil*

O trabalho teve como objetivo segmentar cicatrizes de escorregamentos de terra em áreas distintas da área de treinamento. Foram avaliadas quatro diferentes dimensões de imagens de treinamento (32x32, 64x64, 128x128 e 256x256 pixels) para compreender qual o impacto da dimensão da imagem de treinamento na capacidade de generalização do modelo. A rede U-Net e as imagens RapidEye e o modelo digital de elevação (MDE) da missão SRTM foram utilizados no treinamento dos modelos. A área utilizada para o treinamento corresponde a região de Nova Friburgo (RJ) e as três áreas utilizadas para avaliar os modelos se localizam em Nova Friburgo (RJ), Petrópolis (RJ) e Rolante (RS). Os resultados mostram que os modelos treinados com imagens de maior dimensão tendem a se especializar na detecção dos escorregamentos que estão presentes na área de treino. Assim, os algoritmos treinados com imagens de menor dimensão apresentam melhor capacidade de generalização para as diferentes áreas da área de treinamento, já que os modelos não aprendem padrões específicos da área de treinamento. Os resultados foram pós-processados com diferentes técnicas de processamento de imagens (erosão, dilatação, abertura e fechamento) e notou-se que o pós-processamento dos resultados é uma maneira eficiente para aumentar a precisão dos resultados.

6 CONCLUSÕES

As análises e resultados obtidos nesta pesquisa evidenciam que a técnica de *deep learning* apresenta grande potencial para segmentar de maneira automática os escorregamentos de terra. Porém, em função da complexidade do problema em questão e da variedade de configurações e arquiteturas que a técnica de *deep learning* apresenta, os resultados obtidos evidenciam a importância da dimensão das imagens de treinamento, dos métodos de amostragem e das configurações do hiperparâmetros (*hyperparameters*) dos modelos.

A dimensão das imagens de treinamento é um dos parâmetros mais relevantes, já que de acordo com a natureza do problema que se pretende resolver, ela parece ser um fator limitante. Em problemas em que se deseja segmentar escorregamentos em áreas similares a área de treino, as maiores dimensões analisadas (128 x 128 pixels, 256 x 256 pixels) obtiveram os melhores resultados, pois os modelos tornam-se mais especializados nos dados da área de treinamento. No entanto, em problemas em que se busca segmentar escorregamentos em áreas distintas da área de treinamento, as dimensões menores (32x32 e 64x64 pixels) são mais eficientes uma vez que o modelo aprende padrões mais "locais" dos escorregamentos e da cena e acaba não compreendendo características mais "globais" da cena e específicas da área de treinamento.

O método de amostragem das imagens também é fundamental para buscar reduzir o desbalanceamento entre a classe de interesse (positiva) e a classe de *background* (negativa). O método utilizado se baseou na seleção por localização que se mostrou eficiente para eliminar as áreas que não apresentam a classe positiva e, consequentemente, equilibrar a quantidade de amostras com a classe negativa (mais abundante), permitindo a convergência do modelo.

Os hiperparâmetros da rede como o *batch size*, variações de arquitetura e valor de *learning rate* também são diretamente responsáveis pela acurácia dos resultados obtidos. É fundamental testar diversas combinações de valores entre esses parâmetros, em um processo conhecido como *grid search*, para encontrar a combinação mais eficiente para o problema em questão.

As operações de pós-processamento são eficientes para melhorar a precisão

dos resultados obtidos, porém, não existe uma operação que seja superior a outra, é preciso testar diversas combinações entre elas para encontrar o melhor resultado.

Existem algumas limitações do trabalho realizado que merecem ser ressaltadas para servir de motivação para estudos futuros. As imagens utilizadas no processo de treinamento dos modelos foram apenas do satélite *RapidEye*, portanto, ainda existem dúvidas com relação a capacidade de generalização da técnica de *deep learning* para segmentar os escorregamentos em imagens obtidas por diferentes sensores. Autores como Prakash, Manconi e Loew (2021) e Ghorbanzadeh, Crivellari et al. (2021) realizaram alguns estudos seguindo essa direção, porém existem ainda poucos trabalhos e as conclusões são incipientes para avaliar de maneira ampla o potencial nesse tipo de aplicação. Outro fator que merece atenção é a resolução espacial do modelo digital de elevação (MDE) utilizado nesta pesquisa. Os dados utilizados são da missão SRTM, com 30 metros de resolução espacial, e do SRTM da *Alaska Satellite Facility*, reamostrado para 15 metros de resolução. Assim, uma vez que os dados apresentam resoluções distintas das imagens RapidEye utilizadas (5 metros), em nenhum dos estudos realizados observou-se melhora significativa nos modelos ao utilizar estes MDEs. Os autores Liu et al. (2020), Ghorbanzadeh, Meena et al. (2020) e Meena et al. (2021) observaram melhoras na segmentação utilizando fatores topográficos em conjunto com os dados óticos, porém, ainda não existem trabalhos que avaliem se os dados topográficos também auxiliam na capacidade de generalização dos modelos de *deep learning*.

REFERÊNCIAS

AUGUSTO FILHO, Oswaldo. Caracterização geológico-geotécnica voltada à estabilização de encostas: uma proposta metodológica. In: CONFERÊNCIA Brasileira sobre Estabilidade de Encostas-COBRAE. Anais. [S.l.: s.n.], 1992. p. 721–733.

BLASCHKE, Thomas. Object based image analysis for remote sensing. ISPRS journal of photogrammetry and remote sensing, Elsevier, v. 65, n. 1, p. 2–16, 2010.

BLASCHKE, Thomas et al. Geographic object-based image analysis–towards a new paradigm. ISPRS journal of photogrammetry and remote sensing, Elsevier, v. 87, p. 180–191, 2014.

BUDUMA, Nikhil; LOCASCIO, Nicholas. Fundamentals of deep learning: Designing next-generation machine intelligence algorithms. [S.l.]: "O'Reilly Media, Inc.", 2017.

BURKOV, Andriy. The hundred-page machine learning book. [S.l.]: Andriy Burkov Quebec City, Can., 2019.

CHARU, C Aggarwal. Neural Networks and Deep Learning: A Textbook. [S.l.]: Spinger, 2018.

CHEN, Zhong et al. Automated landslides detection for mountain cities using multi-temporal remote sensing imagery. Sensors, Multidisciplinary Digital Publishing Institute, v. 18, n. 3, p. 821, 2018.

CHOLLET, François. Deep Learning with Python. [S.l.]: Manning, nov. 2017. ISBN 9781617294433.

EM-DAT, CRED. The international disaster database. Center for Research on the Epidemiology of Disasters. Available at: https://www. emdat. be [Accessed on 10/08/2019], 2018.

DIAS, Helen Cristina. Modelagem da suscetibilidade a escorregamentos rasos com base em análises estatísticas. Dissertação (Mestrado em Geografia Física). [S.l.: s.n.], 2019.

DIAS, Helen Cristina et al. Landslide recognition using SVM, Random Forest, and Maximum Likelihood classifiers on high-resolution satellite images: A case study of Itaóca, southeastern Brazil. Brazilian Journal of Geology, SciELO Brasil, v. 51, 2021.

DING, Anzi et al. Automatic recognition of landslide based on CNN and texture change detection. In: IEEE. 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC). [S.l.: s.n.], 2016. p. 444–448.

ESMAEL, Agnaldo Aparecido et al. Classificação multiclasse multiescala de imagens de sensoriamento remoto. [sn], 2015.

GHORBANZADEH, Omid; BLASCHKE, Thomas et al. Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection. Remote Sensing, Multidisciplinary Digital Publishing Institute, v. 11, n. 2, p. 196, 2019.

GHORBANZADEH, Omid; CRIVELLARI, Alessandro et al. A comprehensive transferability evaluation of U-Net and ResU-Net for landslide detection from Sentinel-2 data (case study areas from Taiwan, China, and Japan). Scientific Reports, Nature Publishing Group, v. 11, n. 1, p. 1–20, 2021.

GHORBANZADEH, Omid; MEENA, Sansar Raj et al. Landslide Mapping Using Two Main Deep-Learning Convolution Neural Network Streams Combined by the Dempster–Shafer Model. IEEE Journal of selected topics in applied earth observations and remote sensing, IEEE, v. 14, p. 452–463, 2020.

GHORBANZADEH, Omid; SHAHABI, Hejar et al. Landslide detection using deep learning and object-based image analysis. Landslides, Springer, p. 1–11, 2022.

GONZALEZ, Rafael C.; WOODS, Richard E. Digital image processing. Upper Saddle River, N.J.: Prentice Hall, 2008. ISBN 9780131687288. Disponível em: <http://www.amazon.com/Digital-Image-Processing-3rd-Edition/dp/013168728X>.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. Deep learning. [S.l.]: MIT press, 2016.

HERVÁS, Javier; BOBROWSKY, Peter. Mapping: inventories, susceptibility, hazard and risk. In: LANDSLIDES–DISASTER Risk Reduction. [S.l.]: Springer, 2009. p. 321–349.

HUBEL, David H; WIESEL, Torsten N. Receptive fields of single neurones in the cat's striate cortex. The Journal of physiology, Wiley Online Library, v. 148, n. 3, p. 574–591, 1959.

IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.

JENSEN, John R. Remote sensing of the environment: An earth resource perspective 2/e. [S.l.]: Pearson Education India, 2009.

KELLEHER, John D. Deep learning. [S.l.]: MIT press, 2019.

KINGMA, Diederik P; BA, Jimmy. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

KOBIYAMA, Masato et al. Prevenção de desastres naturais: conceitos básicos. [S.l.: s.n.], 2006.

LECUN, Yann et al. Backpropagation applied to handwritten zip code recognition. Neural computation, MIT Press, v. 1, n. 4, p. 541–551, 1989.

LIU, Peng et al. Research on post-earthquake landslide extraction algorithm based on improved U-Net model. Remote Sensing, Multidisciplinary Digital Publishing Institute, v. 12, n. 5, p. 894, 2020.

LONG, Jonathan; SHELHAMER, Evan; DARRELL, Trevor. Fully convolutional networks for semantic segmentation. In: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [S.l.: s.n.], 2015. p. 3431–3440.

MARR, David. Vision: A computational investigation into the human representation and processing of visual information. CUMINCAD, 1982.

MATHERON, Georges; SERRA, Jean. The birth of mathematical morphology. In: SYDNEY, AUSTRALIA. PROC. 6th Intl. Symp. Mathematical Morphology. [S.l.: s.n.], 2002. p. 1–16.

MEENA, Sansar Raj et al. Rapid mapping of landslides in the Western Ghats (India) triggered by 2018 extreme monsoon rainfall using a deep learning approach. Landslides, Springer, v. 18, n. 5, p. 1937–1950, 2021.

MENESES, Paulo Roberto; ALMEIDA, Tati de;
MELLO BAPTISTA, Gustavo Macedo de. Reflectância dos materiais terrestres. [S.l.]: Oficina de textos, 2019.

MENESES, Paulo Roberto; ALMEIDA, Tati de. Introdução ao processamento de imagens de sensoriamento remoto. Universidade de Brasília, Brasília, 2012.

NG, Andrew et al. Deep learning. CS229 Lecture Notes, p. 1–30, 2014.

PENG, Daifeng; ZHANG, Yongjun; GUAN, Haiyan. End-to-End Change Detection for High Resolution Satellite Images Using Improved UNet++. Remote Sensing, Multidisciplinary Digital Publishing Institute, v. 11, n. 11, p. 1382, 2019.

PRAKASH, Nikhil; MANCONI, Andrea; LOEW, Simon. A new strategy to map landslides with a generalized convolutional neural network. Scientific reports, Nature Publishing Group, v. 11, n. 1, p. 1–15, 2021.

_____. Mapping landslides on EO data: Performance of deep learning models vs. traditional machine learning models. Remote Sensing, Multidisciplinary Digital Publishing Institute, v. 12, n. 3, p. 346, 2020.

QI, Wenwen et al. Automatic mapping of landslides by the ResU-net. Remote Sensing, Multidisciplinary Digital Publishing Institute, v. 12, n. 15, p. 2487, 2020.

QUINN, Joanne et al. Dive into deep learning: Tools for engagement. [S.l.]: Corwin Press, 2019.

RADOVIC, Matija; ADARKWA, Offei; WANG, Qiaosong. Object recognition in aerial images using convolutional neural networks. Journal of Imaging, Multidisciplinary Digital Publishing Institute, v. 3, n. 2, p. 21, 2017.

RONNEBERGER, Olaf; FISCHER, Philipp; BROX, Thomas. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. INTERNATIONAL Conference on Medical image computing and computer-assisted intervention. [S.l.: s.n.], 2015. p. 234–241.

SAMEEN, Maher Ibrahim; PRADHAN, Biswajeet. Landslide Detection Using Residual Networks and the Fusion of Spectral and Topographic Information. IEEE Access, IEEE, v. 7, p. 114363–114373, 2019.

SANTURKAR, Shibani et al. How does batch normalization help optimization? In: ADVANCES in Neural Information Processing Systems. [S.l.: s.n.], 2018. p. 2483–2493.

SHI, Wenzhong et al. Landslide recognition by deep convolutional neural network and change detection. IEEE Transactions on Geoscience and Remote Sensing, IEEE, v. 59, n. 6, p. 4654–4672, 2020.

SRIVASTAVA, Nitish et al. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, JMLR. org, v. 15, n. 1, p. 1929–1958, 2014.

STUMPF, André; KERLE, Norman. Object-oriented mapping of landslides using Random Forests. Remote sensing of environment, Elsevier, v. 115, n. 10, p. 2564–2577, 2011.

TOMINAGA, Lídia Keiko; SANTORO, Jair; AMARAL, Rosangela do. Desastres naturais: conhecer para prevenir. [S.l.]: Instituto geológico, 2009.

UNISDR, UNO. Sendai framework for disaster risk reduction 2015–2030. In: PROCEEDINGS of the 3rd United Nations World Conference on DRR, Sendai, Japan. [S.l.: s.n.], 2015. p. 14–18.

USGS. Landslide types and processes. [S.l.], 2004.

VARNES, David J. Slope movement types and processes. Special report, v. 176, p. 11–33, 1978.

YAN, Le Cun; YOSHUA, B; GEOFFREY, H. Deep learning. nature, v. 521, n. 7553, p. 436–444, 2015.

YI, Yaning; ZHANG, Wanchang. A new deep-learning-based approach for earthquake-triggered landslide detection from single-temporal RapidEye satellite imagery. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, IEEE, v. 13, p. 6166–6176, 2020.

YU, Bo; CHEN, Fang; XU, Chong. Landslide detection based on contour-based deep learning framework in case of national scale of Nepal in 2015. Computers & Geosciences, Elsevier, v. 135, p. 104388, 2020.

ZHONG, Cheng et al. Landslide mapping with remote sensing: challenges and opportunities. International Journal of Remote Sensing, Taylor & Francis, p. 1–27, 2019.

ZHU, Xiao Xiang et al. Deep learning in remote sensing: A comprehensive review and list of resources. IEEE Geoscience and Remote Sensing Magazine, IEEE, v. 5, n. 4, p. 8–36, 2017.

7 APÊNDICE

7.1 Landslide Segmentation with U-Net: Evaluating Different Sampling Methods and Patch Sizes

Landslide Segmentation with U-Net: Evaluating Different Sampling Methods and Patch Sizes

Lucas P. Soares [*a]
Institute of Geosciences
University of São Paulo (USP)
São Paulo, Brazil
lpsoares@usp.br

Helen C. Dias [a]
Institute of Energy and Environment
University of São Paulo (IEE-USP)
São Paulo, 05508-010, Brazil
helen.dias@usp.br

Carlos H. Grohmann [a]
Institute of Energy and Environment
University of São Paulo (IEE-USP)
São Paulo, 05508-010, Brazil
guano@usp.br

July 15, 2020

Abstract

Landslide inventory maps are crucial to validate predictive landslide models; however, since most mapping methods rely on visual interpretation or expert knowledge, detailed inventory maps are still lacking. This study used a fully convolutional deep learning model named U-net to automatically segment landslides in the city of Nova Friburgo, located in the mountainous range of Rio de Janeiro, southeastern Brazil. The objective was to evaluate the impact of patch sizes, sampling methods, and datasets on the overall accuracy of the models. The training data used the optical information from RapidEye satellite, and a digital elevation model (DEM) derived from the L-band sensor of the ALOS satellite. The data was sampled using random and regular grid methods and patched in three sizes (32x32, 64x64, and 128x128 pixels). The models were evaluated on two areas with precision, recall, f1-score, and mean intersect over union (mIoU) metrics. The results show that the models trained with 32x32 tiles tend to have higher recall values due to higher true positive rates; however, they misclassify more background areas as landslides (false positives). Models trained with 128x128 tiles usually achieve higher precision values because they make less false positive errors. In both test areas, DEM and augmentation increased the accuracy of the models. Random sampling helped in model generalization. Models trained with 128x128 random tiles from the data that used the RapidEye image, DEM information, and augmentation achieved the highest f1-score, 0.55 in test area one, and 0.58 in test area two. The results achieved in this study are comparable to other fully convolutional models found in the literature, increasing the knowledge in the area.

Keywords Deep Learning · Fully Convolutional Networks (FCN) · Nova Friburgo · RapidEye · Landslide mapping

1 Introduction

Natural hazards are more frequent and harmful in recent years due to unplanned urbanization, climate change, and population growth [Kobiyama et al., 2006, Hong et al., 2017, Alexander, 2008, Zhong et al., 2019]. According to the Sendai framework for disaster risk reduction 2015-2030 [UNISDR, 2015], between 2008 and 2012, those hazards affected more than 25 million people, with an economic loss of about 1.3 trillion dollars, impeding the progress towards sustainable development.

Landslides commonly cause victims, damages to human habitations, and economic losses. Therefore, the study of landslide detection has been considered a critical area of research in remote sensing [Hong et al., 2017]. However, despite the importance highlighted by many authors, detailed landslides inventories are still lacking [Mondini et al., 2019, Guzzetti et al., 2012]. Landslide inventory maps are used to prepare and validate landslide susceptibility models, evaluate risk and vulnerability, study erosion and geomorphology, and document the impact of a landslide disaster

---
[*]Corresponding Author
[a] Spatial Analysis and Modelling Lab (SPAMLab, IEE-USP) - https://spamlab.github.io

[Van Westen et al., 2008]. Limited and incomplete data may be a source of bias for these studies since model success depends directly on inventory accuracy.

Landslides inventory maps usually are prepared by using remote sensing imagery with high (HR) and very-high (VHR) resolution [Zhong et al., 2020]. The landslides can be recognized in an aerial image manually by visual interpretation, semi-automatically, or automatically by using algorithms for object image analysis (OBIA) and pixel-based classification. Manual classification of landslides is the prevailing method [Xu, 2015, Yu et al., 2020], but, for large areas, it is time-consuming. OBIA methodologies classify landslide areas by grouping objects with similar spectral, spatial, hierarchical, textural, and morphological properties [Blaschke, 2010]. Nevertheless, the assignment of those parameters is highly dependent on the analyst experience. Pixel-based methodologies classify each pixel of the image based on its spectral information. However, geometric and contextual information present in the image is ignored, increasing the salt-and-pepper noise in the results [Stumpf and Kerle, 2011, Blaschke et al., 2014, Zhong et al., 2019, Prakash et al., 2020].

In recent years, deep convolution neural networks (DCNN) achieve state-of-art results in applications such as semantic segmentation, object detection, natural language processing, and speech recognition [Ghorbanzadeh et al., 2019, Peng et al., 2019, Zhu et al., 2017, Long et al., 2015, Radovic et al., 2017]. However, only a few studies have used DCNNs for landslide detection [Zhong et al., 2020].

Ding et al. [2016] used DCNN on GF-1 (Gaofen-1) images with four spectral bands and eight-meter resolution, achieving an overall accuracy of 67%, a detection rate of 72.5%, and 10.2% of false positive rate. Chen et al. [2018] used DCNN on bi-temporal images to evaluate areas with drastic changes and combined a spatial context learning (STCL) and information from a digital elevation model (DEM) to detect landslide areas. The method yield an accuracy of more than 61% on the evaluated areas. Ghorbanzadeh et al. [2019] compared state-of-art machine learning methods and DCNN on RapidEye images and a DEM, with five meters of spatial resolution. The DCNN that used only spectral information and small windows was the best model achieving 78.26% on the mean intersect over union (mIoU) metric. Sameen and Pradhan [2019] compared residual networks (ResNets) trained with topographical information fused using convolutional networks with topographical data added as additional channels. The models trained with the fused data achieved f1-score and mIoU that were superior by 13% and 12.96% compared to the other models. Yu et al. [2020] used the enhanced vegetation index (EVI), DEM degradation indexes, and a contouring algorithm on Landsat images to sample potential landslide zones with less class imbalance distribution. The trained fully convolutional network (PSPNet) achieved 65% of recall and 55.35% of precision. Prakash et al. [2020] used Lidar DEM and Sentinel-2 images to compare traditional pixel, object, and DCNN methods. The deep learning method, U-net with ResNet34 blocks, achieved the best results with the Matthews correlation coefficient score of 0.495 and the probability of detection rate of 0.72.

DCNNs, in supervised learning problems, can learn to identify patterns on the training data without the need for complex operations to extract features or preprocessing methods. However, choosing the best network architecture, preparing the training dataset, and tuning the hyperparameters is still a challenge [Pradhan et al., 2017, Sameen and Pradhan, 2019]. Landslides scars dataset usually have an imbalanced class distribution with more pixels belonging to background objects, such as urban areas, vegetation, and water, than landslide scars [Yu et al., 2020]. Therefore, since landslide scars have different shapes and sizes, sampling methods and patch sizes may affect the model accuracy as it can be a way to reduce the class imbalance between the positive and the negative class.

This research aims to evaluate how different datasets, sampling methods, and patch sizes impacts on the landslide segmentation accuracy of U-net. To achieve that, we trained 288 models with landslide optical information from a RapidEye satellite and topographical information from a DEM derived from the Phased Array type L-band Synthetic Aperture Radar (PALSAR) sensor of the ALOS satellite. The models were trained with images patched in three different sizes (32x32, 64x64, 128x128 pixels), and sampled using random and regular grid sampling methods. Data augmentation was also tested. The study area is in the city of Nova Friburgo, located in the mountainous range of Rio de Janeiro, Brazil. The models were evaluated in two test areas with f1-score, recall, precision, and mean intersect over union (mIoU) metrics.

The main contributions of this research are as follows:

- Broad comparison between patch sizes, sampling method, and datasets.

- Evaluation of U-net architecture for semantic segmentation of landslides.

2 Study Area

In January 2011, an extreme rainfall event (350 mm/48h) triggered at least 3500 translational landslides that killed more than 1500 people and disrupted all major city facilities in the mountainous region of Rio de Janeiro, Brazil. This event is considered the worst Brazilian natural disaster [Avelar et al., 2013].

The mountainous region of Rio de Janeiro encompasses the municipalities of Nova Friburgo, Teresópolis, Petrópolis, Sumidouro, São José do Vale do Rio Preto and Bom Jardim (Fig. 1). The study area is in the municipality of Nova Friburgo, which was severely damaged by the disaster.

Nova Friburgo is in the geomorphological unit of Serra dos Orgãos. The geological units have a WSW-ENE trend, and the elevation ranges between 1100 and 2000 meters above the mean sea level [Dantas, 2001]. The geology consists mainly of igneous and metamorphic rocks such as granites, diorites, gabbros, and gneisses [Tupinambá et al., 2012]. According to Köppen's climate classification scheme [Köppen, 1936], the climate is subtropical highland (Cwb) with dry winter and mild summers. The annual mean precipitation is 1585.62 mm, with most of the rainfall in November, December, and January [Sobral et al., 2018].
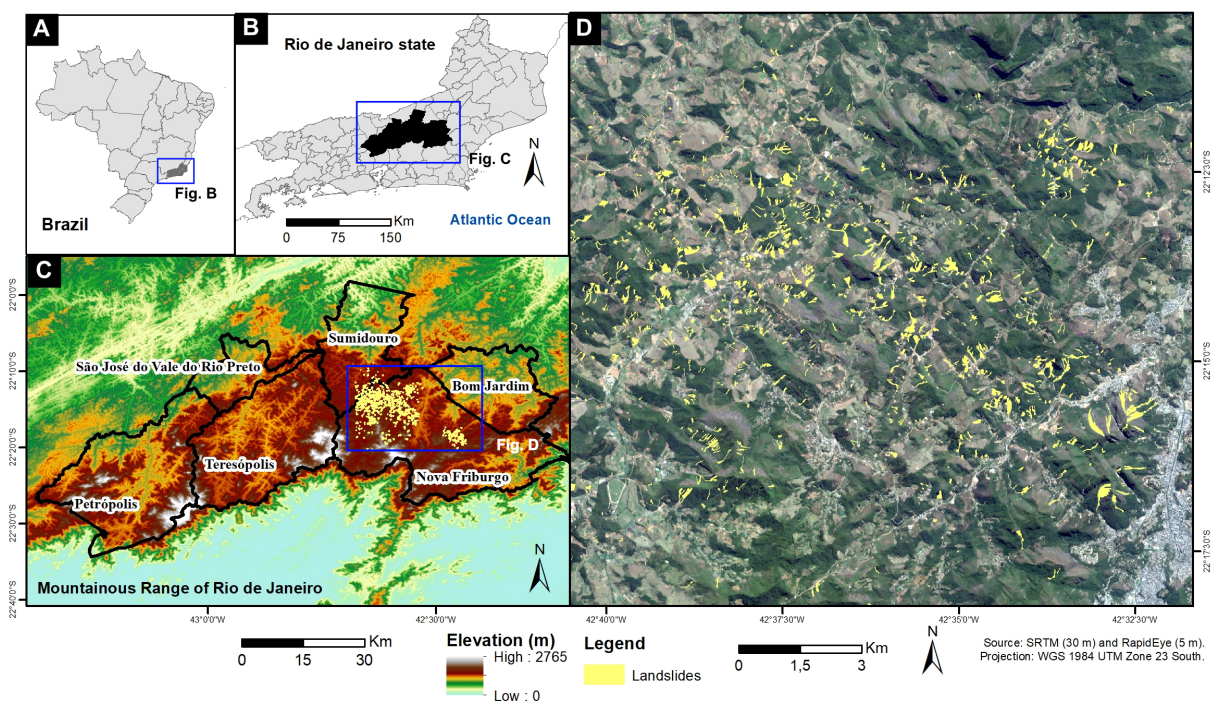
Figure 1: A) Location of the study area in southeastern Brazil. B) Location of the Mountainous Range in the Rio de Janeiro State. C) Mountainous Range of Rio de Janeiro with the study area highlighted in blue. D) True color composition of the RapidEye image over Nova Friburgo, with the segmented landslides in yellow. The image was acquired on 2011-08-13.

3 Methodology

The spectral information from a RapidEye image and topographical information from a digital elevation model (DEM) derived from the ALOS's Phased Array L-band Synthetic Aperture Radar (PALSAR) were used to evaluate the performance of the U-net on landslide segmentation. The models were trained with images in three different window dimensions (32x32, 64x64, 128x128 pixels) that were sampled using random and regular grid methods. Random rotation, vertical, and horizontal flip were used for data augmentation. In total, 288 models were trained (Table 1).

The model's performance was evaluated in two test areas by using the mean intersect over union (mIoU), f1-score, precision, and recall metrics. The proposed methodology involves the following steps: (1) data preprocessing, (2) model training (3) model evaluation.

3

Table 1: Number of trained models on each dataset.

# of Models	Dataset
72	RapidEye
72	RapidEye + Augmentation
72	RapidEye + DEM
72	RapidEye + DEM + Augmentation

3.1 Data Preprocessing

3.1.1 RapidEye

RapidEye consists of a constellation of five identical satellites with high-resolution sensors with a 6.5 meters nominal ground sampling distance at nadir. The orthorectified products are resampled and provided to users at a pixel size of 5 meters. The data are acquired with a temporal resolution of 5 days in five spectral bands: blue (440–510 nm), green (520–590 nm), red (630–685 nm), red-edge (690–730 nm), near-infrared (760–850 nm) [RapidEye, 2011].

This work used the raw digital number (DN) of a 3A product (orthorectified, radiometric, and geometric corrections) with an area of 625 km^2. The image was acquired on 13 August 2011 and downloaded from the Planet Explorer website [Planet Team, 2017].

3.1.2 ALOS/PALSAR

The Phased Array type L-band Synthetic Aperture Radar (PALSAR) is one of the three observation sensors of the Advanced Land Observing Satellite (ALOS). PALSAR data is acquired at an off-nadir angle of 34.3 degrees in a Sun-synchronous Sub-recurrent Orbit (SSO) with a 46-day recurrent period.

In this study, we used the radiometric terrain correction (RTC) product with 12.5 meters of spatial resolution obtained from the Alaska Satellite Facility (ASF) [DAAC, 2015]. The image was acquired on 28 January 2011. The data were resampled to 5 meters pixel resolution with bilinear interpolation method, to match the spatial resolution of the RapidEye image, using GRASS GIS version 7.2.2 [Neteler et al., 2012, GRASS Development Team, 2017].

3.1.3 Data Labeling

The landslides were manually labeled in the RapidEye image using QGIS version 3.8 [QGIS Development Team, 2009]. The segmented landslides were validated with Google Earth Pro version 7.3 [Google, 2019] and by comparison with the landslide map produced by [Netto et al., 2013]. In total, 1007 landslides were extracted from the scene, with area ranging from 200.32 m^2 to 78117.35 m^2 (638.51 m^2 average).

3.1.4 Test Areas

The model's accuracy was evaluated in two test areas with 1024x1024 pixels (Fig. 2). In the first area, agriculture and grazing are predominant, while in the second, native vegetation and human settlements predominate. Ninety-six landslides were extracted from the first area and ninety-one from the second.

3.1.5 Binary Mask

The landslide scars polygons from the train and test areas were rasterized with Rasterio [Gillies et al., 2013–] and Numpy [Oliphant, 2006] Python libraries, to generate a binary mask with the same dimensions of the original scene. The pixels assigned with the value 1 (white pixels) correspond to the landslide scars class and the value 0 (black pixels) to the background class.

3.1.6 Sampling Methods and Patch Sizes

The data was sampled by using random and regular grid methods in three different sizes: 32x32, 64x64, 128x128 pixels (Fig. 3).

The grid method used the bounding coordinates and the image resolution (5 meters) to generate a vector grid. The squares over the grid have an overlap of 20%. The random sampling used the same procedure to generate 5000 sampling square polygons. A select-by-location operation was used to select only the polygons intersecting landslides. This ensures that all sampled images will have at least a small portion of a landslide scar, reducing class imbalance. The code used was adapted from the Keras-Spatial library [Terstriep, 2019].
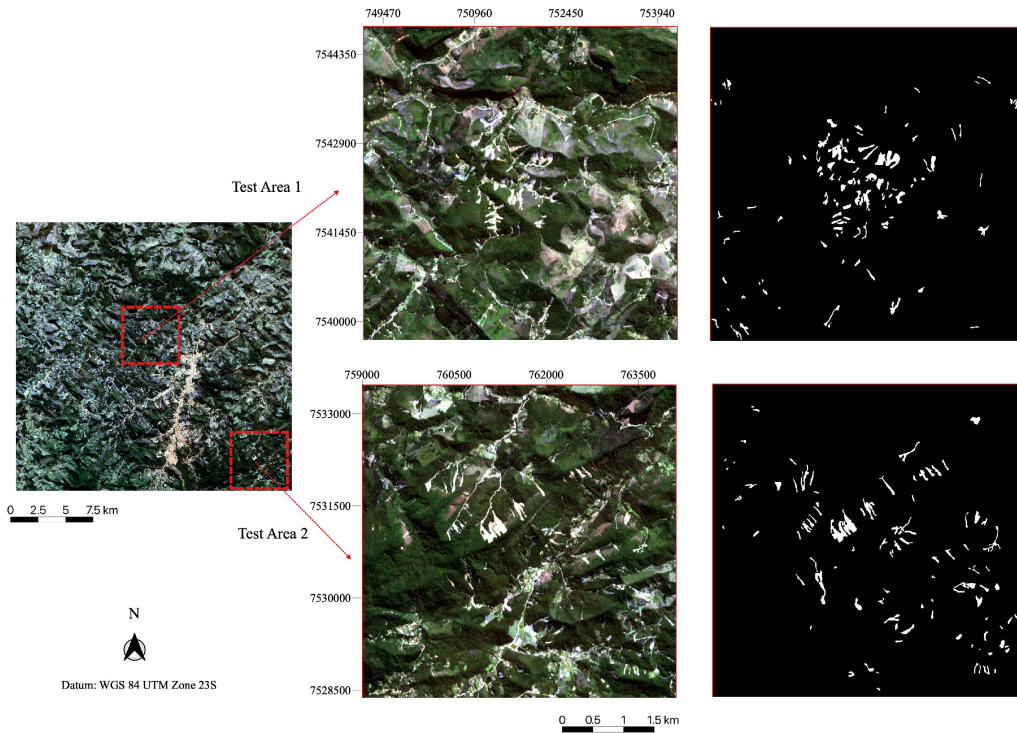
Figure 2: Location of test areas and their binary masks. White pixels represent the manually segmented landslides and black pixels the background.
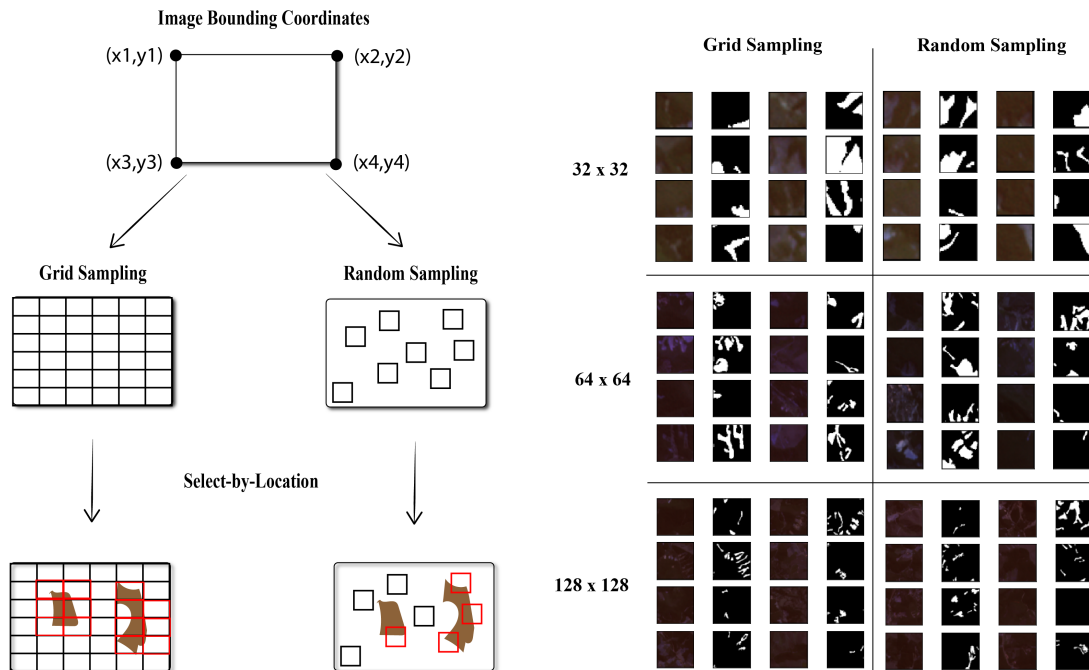

Figure 3: Sampling methods (left) and results obtained for each patch size (right).

3.1.7 Data Augmentation

The Albumentation library [Buslaev et al., 2018] was used to augment the data by using random rotations around $90°$, vertical, and horizontal flips. Table 2 shows the sizes of all the datasets used to train the models.

Table 2: Number of samples from all the datasets and patch sizes used to train the models.

Dataset	Size (pixels)	Regular Sampling	Random Sampling
RapidEye	32x32	3541	1740
	64x64	2264	2368
RapidEye + DEM	128x128	1565	3653
RapidEye + Augmentation	32x32	9912	4872
	64x64	6336	6628
RapidEye + DEM + Augmentation	128x128	4380	10228

3.2 Model Training

3.2.1 Model Architecture

U-net [Ronneberger et al., 2015] is a fully convolutional network developed for the segmentation of biomedical images. This type of architecture does not use fully connected layers in their structure; instead, they have an encoder-decoder architecture with just convolutional layers. The encoder path is responsible for classifying the pixels, but without taking the spatial location into account. The decoder path uses up-convolutions and concatenation to recover the spatial location of the classified pixels and return a mask with the same dimensions of the input image.

In this study, we evaluated the U-net architecture (Fig. 4) in three different values of initial filters: 16, 32, and 64 filters. The convolutional blocks on the encoder path have two 3x3 convolutional layers, activated with ReLu non-linear function, and followed by a max-pooling operation that reduces the spatial dimension by 2. In each convolutional block, the number of filters increases by 2^n, where n is the block's position. On the decoder path, 2x2 up-sampling operations increase the data's spatial dimension to allow the concatenation of feature maps with the same dimension from the encoder path. Then, the concatenated data serve as input for two convolutional layers before another up-sampling operation.
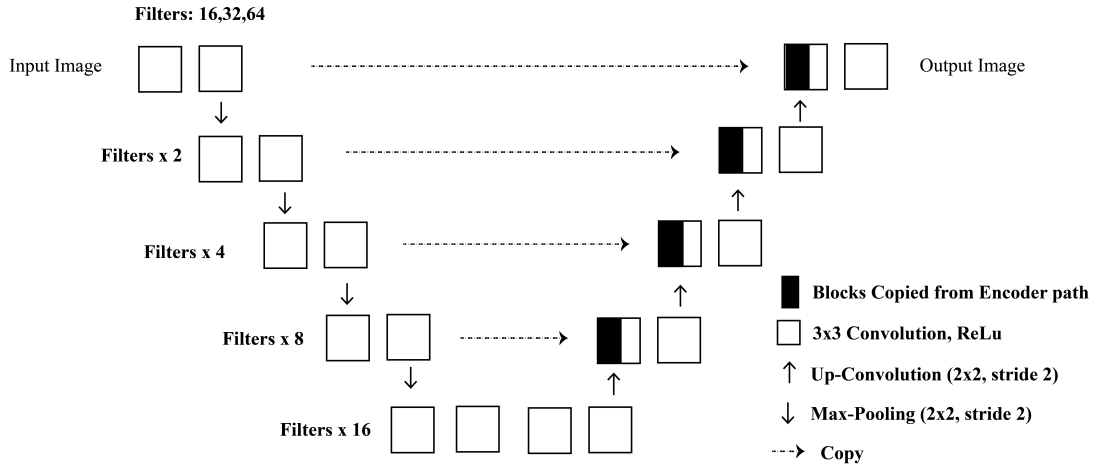
Figure 4: U-net architecture. Filters increase by 2^n in each convolutional block.

3.2.2 Hyperparameters

The models were trained for 200 epochs with a fixed learning rate of 0.001. The initial tests also evaluated 0.01 and 0.0001 learning rate values, but the model's accuracy was lower than the models trained with a learning rate of 0.001. Binary cross-entropy and Adam were used as the loss and optimization function, respectively. The models were trained with four different batch sizes (16, 32, 64, 128 samples). The model's weights were just saved when the validation loss function decrease to reduce the overfitting.

The models were trained on Google Colaboratory virtual environment [Google, 2018] with Keras [Chollet et al., 2015] and Tensorflow [Abadi et al., 2015] Python libraries. 30% of each dataset was used as validation data.

3.3 Evaluation Metrics

The model's performance was evaluated over two test areas by using f1-score, recall, precision, and mean intersect over union (mIoU) metrics. Those metrics are based on true positives (TP), false positives (FP), and false negatives (FN). TP are pixels correctly classified as landslides. FP represents the pixels incorrectly classified as landslides, and FN the pixels incorrectly classified as the background [Ghorbanzadeh et al., 2019, 2018, Guirado et al., 2017]. The models that were trained with DEM as an additional channel were evaluated on test areas with an additional DEM channel.

3.3.1 Precision

Precision (Eq. 1) defines how accurate the model is by evaluating how much of the classified areas are landslides. The metric is useful for evaluating the cost of false positives.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{1}$$

3.3.2 Recall

Recall (Eq. 2) calculates how many of the actual positives are true positives. This metric is suitable to evaluate the cost associated with false negatives.

$$Recal = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{2}$$

3.3.3 F1-Score

F1-score (Eq. 3) combines precision and recall to measure if there is a balance between true positives and false negatives.

$$F1 - Score = 2 * \frac{Precision\ *\ Recall}{Precision + Recall} \tag{3}$$

3.3.4 Mean Intersect Over Union (mIoU)

Mean intersect over union (Eq. 4), also known as Jaccard Index, computes the overlapping of areas between the ground truth (A) and the model prediction (B) divided by the union of these areas. Then, the values are averaged for each class. A value of 1 (one) represents perfect overlapping, while 0 (zero) represents no overlap.

$$mIoU = \frac{A \cap B}{A \cup B} = \frac{True\ Positives}{True\ Positives + False\ Positives + False\ Negatives} \tag{4}$$

The result section shows for each dataset, sampling method, and patch size the models with the highest F1-Score and mIoU. The complete results are available in the Supplementary Material. The model generalization was evaluated by averaging the mIoU values from both test areas.

4 Results and Discussion

The models were evaluated on two test areas with precision, recall, f1-score, and mIoU metrics. The results (Fig. 5) shows that the models trained with the RapidEye+DEM and RapidEye+DEM+Augmentation datasets achieved the best results in all evaluated metrics in test area one. The models trained with 32x32 tiles had the lowest precision (0.24) over all the datasets, while models trained with regular 64x64 and random 128x128 tiles from the RapidEye+DEM dataset achieved 0.67 and 0.66 of precision. The recall was higher for the models trained with 128x128 regular tiles (0.68) and 32x32 random tiles (0.65). The model trained with 128x128 random tiles from the RapidEye+DEM+Augmentation dataset achieved the best f1-score (0.55) and mIoU (0.38).
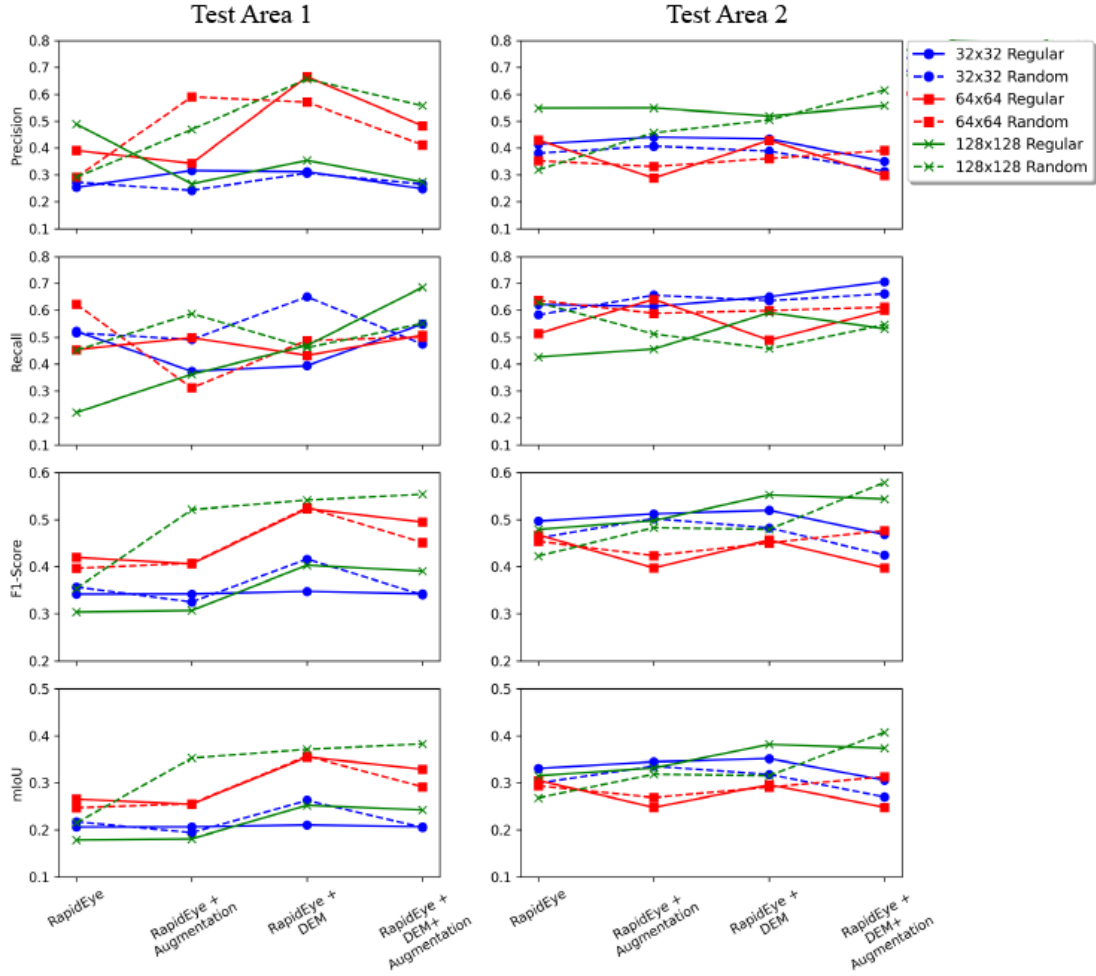
Figure 5: Precision, recall, f1-score and mIoU of the models with the highest f1-score and mIoU in test areas one (left) and two (right).

In test area two, the dataset had a smaller influence over the results; however, models trained with RapidEye+DEM+ +Augmentation dataset also achieved the best results. The model trained with 128x128 random tiles achieved the highest precision (0.62), f1-score (0.58), and mIoU (0.41). Just recall was higher for the 32x32 regular tiles (0.70).

Precision evaluates the cost of false positives, while recall evaluates the cost of false negatives. At test area one (Fig. 6, left), the 32x32 models had lower precision meaning that they misclassify more background areas as landslides (false positives). Similar results occur at test area two (Fig. 6, right), but 64x64 models also had low precision values. Recall varied among the datasets; nevertheless, in both test areas, the models trained with 32x32 tiles achieved high results. Therefore, these models classified more landslide areas as landslides (true positive), reducing false negatives.

The F1-score value will always show a positive correlation with mIoU; however, mIoU tends to penalize incorrect classifications more quantitatively than f1-score. In both test areas, the models trained with random 128x128 tiles from the RapidEye+DEM+Augmentation dataset had the best f1-score and mIoU. In the test area one, this model achieved 0.55 of f1-score and 0.38 of mIoU. While in test area two, it achieved an f1-score of 0.58 and mIoU of 0.41. Comparing with the RapidEye dataset, where these models had the worst performance, the f1-score and mIoU increased 0.2 and 0.16 in test area one, and 0.16, 0.14 in test area two.

When the test areas are evaluated individually, the sampling method seems to be less critical than the dataset to the overall accuracy of the models. However, by averaging the mIoU score of both test areas, it can be seen (Table 3) that random sampling outperformed the regular sampling. Thus, random sampling helped in increasing the generalization
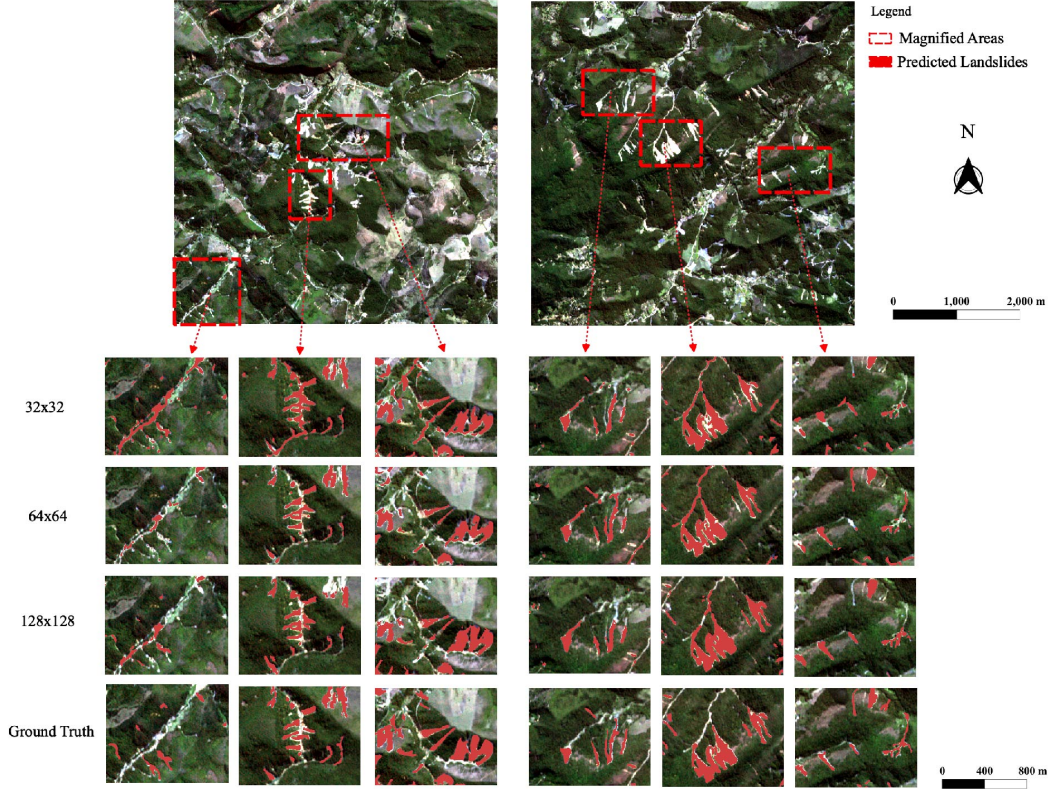
Figure 6: Results of the models with the highest mIoU on each test area. In test area one (left), the image shows the models trained with 32x32 and 64x64 regular tiles from the RapidEye+DEM dataset, and 128x128 random tiles from the RapidEye+DEM+Augmentation dataset. In test are two (right), the image shows the models trained 32x32 and 64x64 regular tiles from RapidEye+DEM dataset, and the 128x128 model trained with random tiles from the RapidEye+DEM+Augmentation dataset.

capacity of the models. Moreover, similar to what was observed in each test area individually, models trained with DEM and DEM+Augmentation had a better performance.

Table 3: Results of the models with the highest average mIoU.

Sampling	Size	Test Area 1 - mIoU	Test Area 2 - mIoU	Average mIoU	Dataset
Random	32	0.26	0.32	0.29	RapidEye+DEM
Random	64	0.29	0.31	0.30	RapidEye+DEM+Augmentation
Random	128	0.31	0.41	0.36	RapidEye+DEM+Augmentation

The results (Fig. 7) of each model from table 3, shows that the 32x32 model predicted, in both test areas, 0.36 and 0.38 km^2 of true positives, 0.82 and 0.59 km^2 of false positives, achieving the highest values. While the model trained with 128x128 tiles predicted the smallest true positive areas (0.26 and 0.32 km^2), false positive areas (0.29 and 0.20 km^2), and larger false negatives (0.30 and 0.27 km^2) and true negatives (25.37 and 25.42 m^2) areas. The model trained with 64x64 tiles achieved values in between those two models, with 0.29 and 0.36 km^2 of true positives; 0.46, 0.57 km^2 of false positives; 0.27 and 0.23 km^2 of false negatives, and true negative of 25.19 and 25.05 km^2.

The results of all evaluated models suggest that the models trained with smaller window sizes tend to understand the local context better. Thus, they classify more landslides correctly, achieving higher true positives and lower false negative values. However, as they are trained with small tiles, the scene's global context, which helps differentiate the background areas, is lost. As a result, they misclassify more background areas as landslides (false positives). On the other hand, models trained with the larger window sizes, in general, understand the global context better, reducing

9

32x32	Test Area 1	Test Area 2
TP	0.36	0.38
FP	0.82	0.59
FN	0.19	0.22
TN	24.83	25.03

64x64	Test Area 1	Test Area 2
TP	0.29	0.36
FP	0.46	0.57
FN	0.27	0.23
TN	25.19	25.05

128x128	Test Area 1	Test Area 2
TP	0.26	0.32
FP	0.29	0.20
FN	0.30	0.27
TN	25.37	25.42

TP = True Positive (km^2)
FP = False Positive (km^2)
FN = False Negative (km^2)
TN = True Negative (km^2)

Datum: WGS 1984 UTM Zone 23South
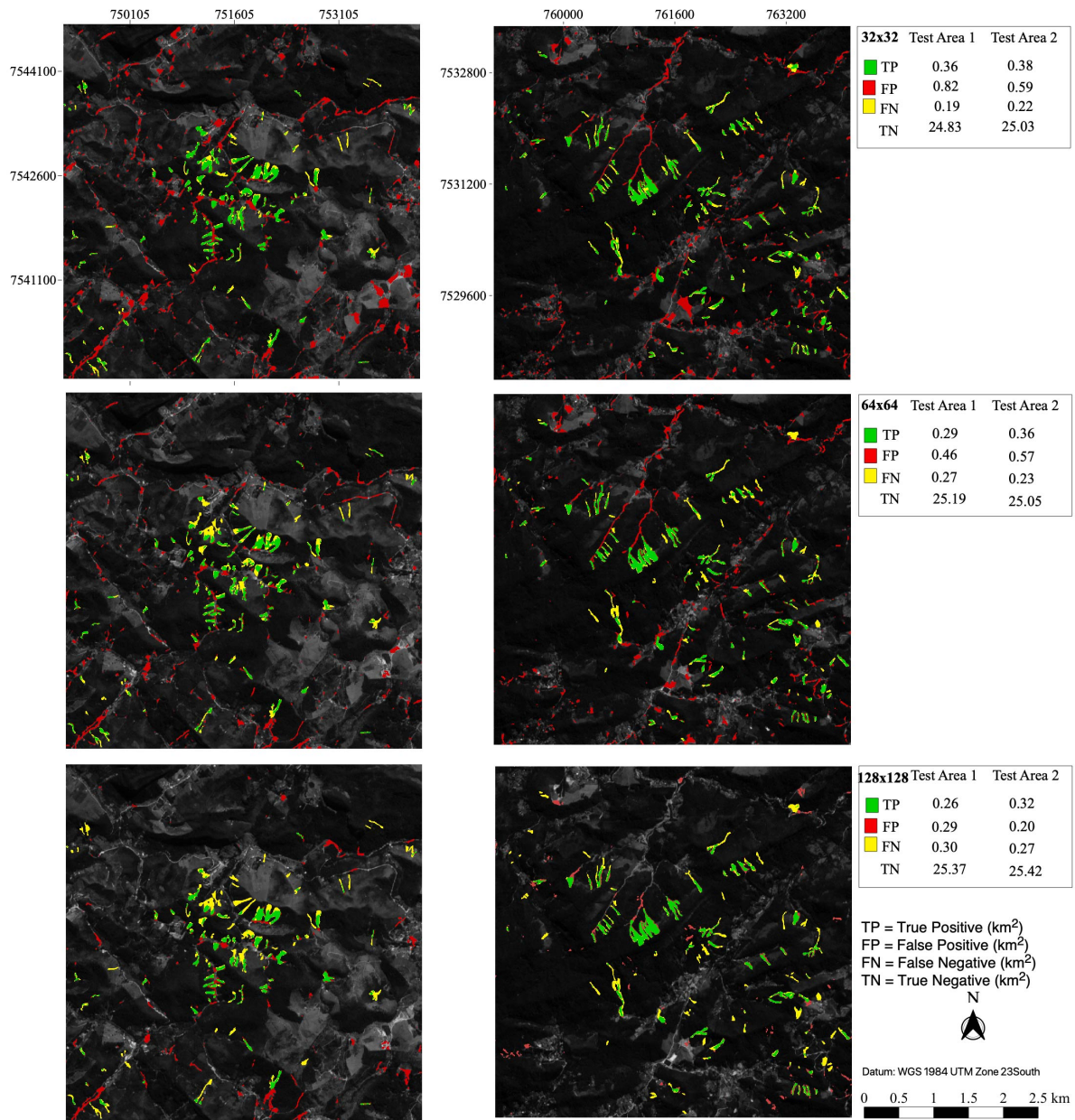
Figure 7: Results of the models with the best generalization results (Table 3), for each patch size, on test area one (left) and two (right).

the false positive errors. Nevertheless, they classify a smaller number of pixels representing landslide scars correctly, increasing the number of false negatives.

Areas with similar spectral responses to landslides such as rivers with increased bedload, gravel roads, grazing, and agricultural areas are more common in area one than area two. Therefore, the models usually make more false positives errors in this area. In area two, the most common false positive mistakes were due to human settlements.

5 Conclusions

This study's main objective was to assess how the datasets, sampling methods, and patch sizes impact the overall performance of U-net on landslide segmentation. Our study suggests that the use of DEM and augmentation helped increase the overall accuracy of the models. Random sampling helped in increasing model generalization. Models trained with 32x32 patches classified more landslides areas correctly, thus, achieving higher true positive areas and lower false negatives. However, they also predict more false positive areas directly impacting precision, f1-score, and mIoU values. Contrarily, models trained with 128x128 tiles make less false positive errors and predict more areas correctly as background. Nevertheless, they also misclassify more landslide areas, increasing the number of false negatives and reducing the recall value. 64x64 tile models achieved results that lie in between the 32x32 and 128x128 models.

In our study, the use of the digital elevation model as an additional channel helped improve the accuracy of the models. This results differs from the ones obtained by Sameen and Pradhan [2019] and Ghorbanzadeh et al. [2019]. However, since both authors used DEMs with higher spatial resolutions and the models were trained with different architectures and smaller tiles than the ones used in this research, more study is needed to address the most effective ways to use DEM with deep learning models. The 128x128 random model trained with the RapidEye+DEM+Augmentation dataset achieved the best performance in this research. The F1-score, 0.55 and 0.58, achieved in both test areas, is comparable to U-Net + ResNet34 evaluated by Prakash et al. [2020], which achieved 0.56 of f1-score, and the PspNet tested by Yu et al. [2020] that achieved f1-score of 0.6. Future studies should explore multi-input models that can be trained with different input sizes; and evaluate different post-processing segmentation techniques to increase the quality of the results.

Computer Code Availability

All the codes used in this research are available on GitHub: `https://github.com/lpsmlgeobr/Landslide_segmentation_with_unet`.

Data Availability

The RapidEye image used in this study (Image ID:2328825) was acquired from Planet (`www.planet.com`) through Planet's Education and Research Program. The ALOS PALSAR DEM (Tile 26708) is available from the Alaska Satellite Facility (ASF) Distributed Active Archive Center (DAAC – `https://search.asf.alaska.edu/`).

Acknowledgements

References

Masato Kobiyama, Magaly Mendonça, Davis Anderson Moreno, IPVO Marcelino, Emerson V Marcelino, Edison F Gonçalves, Leticia LP Brazetti, Roberto F Goerl, Gustavo SF Molleri, and F de M Rudorff. *Prevenção de desastres naturais: conceitos básicos*. Organic Trading Curitiba, 2006.

Haoyuan Hong, Wei Chen, Chong Xu, Ahmed M Youssef, Biswajeet Pradhan, and Dieu Tien Bui. Rainfall-induced landslide susceptibility assessment at the chongren area (china) using frequency ratio, certainty factor, and index of entropy. *Geocarto international*, 32(2):139–154, 2017.

David E Alexander. A brief survey of gis in mass-movement studies, with reflections on theory and methods. *Geomorphology*, 94(3-4):261–267, 2008.

Cheng Zhong, Yue Liu, Peng Gao, Wenlong Chen, Hui Li, Yong Hou, Tuohuti Nuremanguli, and Haijian Ma. Landslide mapping with remote sensing: challenges and opportunities. *International Journal of Remote Sensing*, pages 1–27, 2019.

UNO UNISDR. Sendai framework for disaster risk reduction 2015–2030. In *Proceedings of the 3rd United Nations World Conference on DRR, Sendai, Japan*, pages 14–18, 2015.

Alessandro C Mondini, Michele Santangelo, Margherita Rocchetti, Enrica Rossetto, Andrea Manconi, and Oriol Monserrat. Sentinel-1 sar amplitude imagery for rapid landslide detection. *Remote Sensing*, 11(7):760, 2019.

Fausto Guzzetti, Alessandro Cesare Mondini, Mauro Cardinali, Federica Fiorucci, Michele Santangelo, and Kang-Tsung Chang. Landslide inventory maps: New tools for an old problem. *Earth-Science Reviews*, 112(1-2):42–66, 2012.

Cees J Van Westen, Enrique Castellanos, and Sekhar L Kuriakose. Spatial data for landslide susceptibility, hazard, and vulnerability assessment: an overview. *Engineering geology*, 102(3-4):112–131, 2008.

Cheng Zhong, Yue Liu, Peng Gao, Wenlong Chen, Hui Li, Yong Hou, Tuohuti Nuremanguli, and Haijian Ma. Landslide mapping with remote sensing: challenges and opportunities. *International Journal of Remote Sensing*, 41(4):1555–1581, 2020.

Chong Xu. Preparation of earthquake-triggered landslide inventory maps using remote sensing and gis technologies: Principles and case studies. *Geoscience Frontiers*, 6(6):825–836, 2015.

Bo Yu, Fang Chen, and Chong Xu. Landslide detection based on contour-based deep learning framework in case of national scale of nepal in 2015. *Computers & Geosciences*, 135:104388, 2020.

Thomas Blaschke. Object based image analysis for remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 65(1):2–16, 2010.

André Stumpf and Norman Kerle. Object-oriented mapping of landslides using random forests. *Remote sensing of environment*, 115(10):2564–2577, 2011.

Thomas Blaschke, Geoffrey J Hay, 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*, 87:180–191, 2014.

Nikhil Prakash, Andrea Manconi, and Simon Loew. Mapping landslides on eo data: Performance of deep learning models vs. traditional machine learning models. *Remote Sensing*, 12(3):346, 2020.

Omid Ghorbanzadeh, Thomas Blaschke, Khalil Gholamnia, Sansar Raj Meena, Dirk Tiede, and Jagannath Aryal. Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection. *Remote Sensing*, 11(2):196, 2019.

Daifeng Peng, Yongjun Zhang, and Haiyan Guan. End-to-end change detection for high resolution satellite images using improved unet++. *Remote Sensing*, 11(11):1382, 2019.

Xiao Xiang Zhu, Devis Tuia, Lichao Mou, Gui-Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36, 2017.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

Matija Radovic, Offei Adarkwa, and Qiaosong Wang. Object recognition in aerial images using convolutional neural networks. *Journal of Imaging*, 3(2):21, 2017.

Anzi Ding, Qingyong Zhang, Xinmin Zhou, and Bicheng Dai. Automatic recognition of landslide based on cnn and texture change detection. In *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 444–448. IEEE, 2016.

Zhong Chen, Yifei Zhang, Chao Ouyang, Feng Zhang, and Jie Ma. Automated landslides detection for mountain cities using multi-temporal remote sensing imagery. *Sensors*, 18(3):821, 2018.

Maher Ibrahim Sameen and Biswajeet Pradhan. Landslide detection using residual networks and the fusion of spectral and topographic information. *IEEE Access*, 7:114363–114373, 2019.

Biswajeet Pradhan, Maher Ibrahim Seeni, and Haleh Nampak. Integration of lidar and quickbird data for automatic landslide detection using object-based analysis and random forests. In *Laser Scanning Applications in Landslide Assessment*, pages 69–81. Springer, 2017.

André S Avelar, Ana L Coelho Netto, Willy A Lacerda, Leonardo B Becker, and Marcos B Mendonça. Mechanisms of the recent catastrophic landslides in the mountainous range of rio de janeiro, brazil. In *Landslide science and practice*, pages 265–270. Springer, 2013.

M. E. Dantas. Geomorfologia do estado do rio de janeiro. *CPRM. Estudo geoambiental do Estado do Rio de Janeiro. Brasília*, 2001.

Miguel Tupinambá, Monica Heilbron, Beatriz Paschoal Duarte, Julio Cesar Horta de Almeida, Claudia Sayão Valladares, Bruno Trota Pacheco, Marcelo dos Santos Salomão, Flávio Ribeiro Conceição, Luiz Guilherme Eirado da Silva, Clayton Guia de Almeida, Marcelo Ambrósio Ferrassoli, Mariana de Cássia O. da Costa, Luciana Rocha Tupinambá,

David Silva Rocha, Pedro Monteiro Benac, Hugo Mathias O. Carvalho da Silva, Paulo Vicente Guimarães, (DRM-RJ), (DRM-RJ), Felipe de Lima da Silva, Nely Palermo, and Ronaldo Mello Pereira. Mapa geológico folha nova friburgo sf-23-z-b-ii. Technical report, CPRM - Serviço Geológico do Brasil, 2012.

W. Köppen. *Das geographische System der Klimate*, volume 1, chapter Das geographische System der Klimate, pages 1–44. Gerbrüder Bornträger, 1936.

Bruno Serafini Sobral, José Francisco Oliveira-Júnior, Givanildo Gois, Paulo Miguel de Bodas Terassi, and João Gualberto Rodrigues Muniz-Júnior. Variabilidade espaço-temporal e interanual da chuva no estado do rio de janeiro. *Revista Brasileira de Climatologia*, 22, 2018.

AG RapidEye. Satellite imagery product specifications. *Satellite imagery product specifications: Version*, 2011.

Planet Team. Planet application program interface: In space for life on earth. san francisco, ca, 2017. Available online: `https://api.planet.com`. Last accessed on 2020-05-1.

ASF DAAC. Alos palsar_radiometric_terrain_corrected_high_res. *JAXA/METI*, 11, 2015. `https://search.asf.alaska.edu/#/`, Last accessed on 2019-12-05.

Markus Neteler, M. Hamish Bowman, Martin Landa, and Markus Metz. GRASS GIS: A multi-purpose open source GIS. *Environmental Modelling & Software*, 31:124–130, 2012. ISSN 1364-8152. doi: 10.1016/j.envsoft.2011.11.014.

GRASS Development Team. *Geographic Resources Analysis Support System (GRASS GIS) Software, Version 7.2*. Open Source Geospatial Foundation, 2017. URL `http://grass.osgeo.org`.

QGIS Development Team. *QGIS Geographic Information System*. Open Source Geospatial Foundation, 2009. URL `http://qgis.osgeo.org`.

Google. Google earth pro version 7.3, 2019. `https://www.google.com/earth/versions/#download-pro`, Last accessed on 2020-04-20.

Ana Luiza Coelho Netto, Anderson Mululo Sato, André de Souza Avelar, Lílian Gabriela G Vianna, Ingrid S Araújo, David LC Ferreira, Pedro H Lima, Ana Paula A Silva, and Roberta P Silva. January 2011: the extreme landslide disaster in brazil. In *Landslide science and practice*, pages 377–384. Springer, 2013.

Sean Gillies et al. Rasterio: geospatial raster i/o for Python programmers, 2013–. URL `https://github.com/mapbox/rasterio`.

Travis E Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006.

Jeff Terstriep. Keras spatial, 2019. `https://pypi.org/project/keras-spatial/`, Last accessed on 2020-02-10.

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

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Google. Google colaboratory, 2018. `https://colab.research.google.com/`, Last accessed on 2020-05-12.

François Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015.

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, 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 Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Omid Ghorbanzadeh, Dirk Tiede, Zahra Dabiri, Martin Sudmanns, and Stefan Lang. Dwelling extraction in refugee camps using cnn–first experiences and lessons learnt. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 2018.

Emilio Guirado, Siham Tabik, Domingo Alcaraz-Segura, Javier Cabello, and Francisco Herrera. Deep-learning convolutional neural networks for scattered shrub detection with google earth imagery. *arXiv preprint arXiv:1706.00917*, 2017.

7.2 Landslide Segmentation with Deep Learning: Evaluating Model Generalization in Rainfall-Induced Landslides in Brazil

Article

Landslide Segmentation with Deep Learning: Evaluating Model Generalization in Rainfall-Induced Landslides in Brazil

Lucas Pedrosa Soares [1,2] , Helen Cristina Dias [2] , Guilherme Pereira Bento Garcia [1,2] and Carlos Henrique Grohmann [2,*]

1   Institute of Geosciences, University of São Paulo (IGc-USP), São Paulo 05508-080, Brazil; lpsoares@usp.br (L.P.S.); guilherme.pereira.garcia@usp.br (G.P.B.G.)
2   Spatial Analysis and Modelling Lab (SPAMLab), Institute of Energy and Environment, University of São Paulo (IEE-USP), São Paulo 05508-080, Brazil; helen.dias@usp.br
*   Correspondence: guano@usp.br

Abstract: Automatic landslide mapping is crucial for a fast response in a disaster scenario and improving landslide susceptibility models. Recent studies highlighted the potential of deep learning methods for automatic landslide segmentation. However, only a few works discuss the generalization capacity of these models to segment landslides in areas that differ from the ones used to train the models. In this study, we evaluated three different locations to assess the generalization capacity of these models in areas with similar and different environmental aspects. The model training consisted of three distinct datasets created with RapidEye satellite images, Normalized Vegetation Index (NDVI), and a digital elevation model (DEM). Here, we show that larger patch sizes (128×128 and 256×256 pixels) favor the detection of landslides in areas similar to the training area, while models trained with smaller patch sizes (32×32 and 64×64 pixels) are better for landslide detection in areas with different environmental aspects. In addition, we found that the NDVI layer helped to balance the model's results and that morphological post-processing operations are efficient for improving the segmentation precision results. Our research highlights the potential of deep learning models for segmenting landslides in different areas and is a starting point for more sophisticated investigations that evaluate model generalization in images from various sensors and resolutions.

Keywords: deep learning; landslides; U-Net; automatic segmentation

1. Introduction

Landslides are one of the most frequent and destructive natural hazards worldwide. They are responsible for causing infrastructure damages, economic losses, and victims, mainly when it occurs near human habitation [1–3]. In recent years, increased deforestation, unplanned urbanization, climate change, and population growth have enhanced the impact of these events on human lives and infrastructure [4–8]. In 2021, according to the Emergency Event Database (EM-DAT), landslides were classified as the second most costly disaster and caused 40 billion dollars of economic losses in Germany alone and 234 deaths in India [9].

In South America, Brazil concentrates around 40% of all fatal landslides in the continent [2]; several events that occurred in the past few decades in the country led to social and economical losses [10–12]. Therefore, landslide detection studies have been considered critical in remote sensing [5]. However, despite the importance highlighted by many authors, detailed landslide inventories are still scarce [13–15]. Asia/Oceania and Europe lead the publication of studies about landslide inventory construction [16–19]. Nevertheless, several countries, such as Brazil, lack common procedures to recognize landslide features on the landscape [20]. Landslide inventory maps are used to prepare and validate landslide susceptibility models [16,21–24], evaluate risk and vulnerability [25–31], perform geormorphometric (geomorphology) studies [29,32–39], and evaluate landslide events [40]. Limited

and incomplete data may be a source of bias for these studies since model success depends directly on inventory accuracy [41,42].

Landslide inventory maps are usually prepared using high (HR) or very high (VHR) resolution remote sensing imagery [7]. Detection of landslides can be performed manually by aerial image visual interpretation [43–49], semi-automatically, or automatically by using object-based image analysis (OBIA) algorithms [49–51] and pixel-based classification [52,53]. Manual classification of landslides is the prevailing method [14,54,55], but is costly, exhaustive, and time-consuming, almost impracticable for large areas. OBIA is an alternative method for HR and VHR image analysis. The method is based on objects rather than individual pixels [56]. Object-based approaches have two main steps: segmentation and classification [56,57]. Subsequently expert knowledge can be added to the analysis. After segmentation, several object characteristics can be used to classify landslide areas, such as spectral, spatial, hierarchical, textural, and morphological [56]. Pixel-based methods classify each pixel of the image based on its spectral information, ignoring geometric and contextual information, which increases the salt-and-pepper noise in the results [58–60].

In recent years, deep convolutional neural networks (DCNN) have achieved state-of-the-art results in applications such as semantic segmentation, object detection, natural language processing, and speech recognition [61–65]. However, only a few studies have used DCNNs for landslide detection [7].

The recent literature covers topics that evaluate how different architectures affect the model accuracy; the impact of patch size, sampling, and different layers in the results, and the generalization capacity of deep learning models to detect landslides in different areas. Sameen and Pradhan [66] compared residual networks (ResNets) trained with topographical information fused by convolutional networks with topographical data added as additional channels. The models trained with the fused data achieved f1 score and mean intersection over union (mIoU) that were superior by 13% and 12.96% compared to the other models. Ghorbanzadeh et al. [61] compared state-of-the-art machine learning methods and DCNN using RapidEye images and a DEM, with five meters of spatial resolution. The DCNN that used only spectral information and small windows was the best model, achieving 78.26% on the mIoU metric. Yi and Zhang [67] evaluated the LandsNet architecture in two test areas with different environmental characteristics. The results were optimized with morphological operations and the proposed approach yielded an f1 score of 86.89%. Yu et al. [55] used the enhanced vegetation index (EVI), DEM degradation indexes, and a contouring algorithm on Landsat images to sample potential landslide zones with less class imbalance distribution. The trained fully convolutional network (PSPNet) achieved 65% of recall and 55.35% of precision. Prakash et al. [60] used lidar DEM and Sentinel-2 images to compare traditional pixel-based, object-based, and DCNN methods. The deep learning method, U-net with ResNet34 blocks, achieved the best results, with the Matthews correlation coefficient score of 0.495 and the probability of detection rate of 0.72. Prakash et al. [68] evaluated a U-Net in a progressive training with different image spatial resolutions and sensors that used a combination of landslide inventories to predict landslides in different locations around the world. The highest Matthews correlation coefficient achieved was 0.69.

DCNNs, in supervised learning problems, can learn to identify patterns on the training data without the need for complex operations to extract features or pre-processing methods. However, choosing the best network architecture, preparing the training dataset, and tuning the hyperparameters is still a challenge [66,69]. Landslide scar datasets usually have an imbalanced class distribution, with more pixels belonging to background objects, such as urban areas, vegetation, and water, than landslide scars [55]. Therefore, since landslide scars have different shapes and sizes, sampling methods and patch sizes may affect the model accuracy as it can be a way to reduce the class imbalance between the positive and the negative class. Moreover, to the best of our knowledge, only Prakash et al. [68] evaluated the generalization capacity of deep learning models. However, the scenes used to evaluate the models usually are in vegetated areas, where the contrast between the landslide scars

and vegetation allows the models to distinguish the landslides. Moreover, only Yi and Zhang [67] tested post-processing operations to improve the segmentation results.

Thus, the objective of this study is to evaluate model generalization and post-processing techniques with models trained with different datasets and patch sizes in scenes with varying spatial complexity. The main contribution of this paper is as follows:

- Evaluation of model generalization in areas with different scene complexity in Brazil;
- Evaluation of binary opening, closing, dilation, and erosion as post-processing techniques;
- Evaluation of how different patch sizes affect model generalization;
- Evaluation of different datasets on model generalization.

2. Study Areas

The study areas (Figure 1) were located in Rio de Janeiro (RJ) and Rio Grande do Sul (RS) states in the southern part of Brazil. The areas located in the city of Nova Friburgo (RJ state) were used to train the deep learning models and were considered as test area 1 (TA1). The area close to the city of Teresópolis, which is also located in RJ state, was used as test area 2 (TA2); and test area 3 (TA3) was located close to the city of Rolante (RS state).
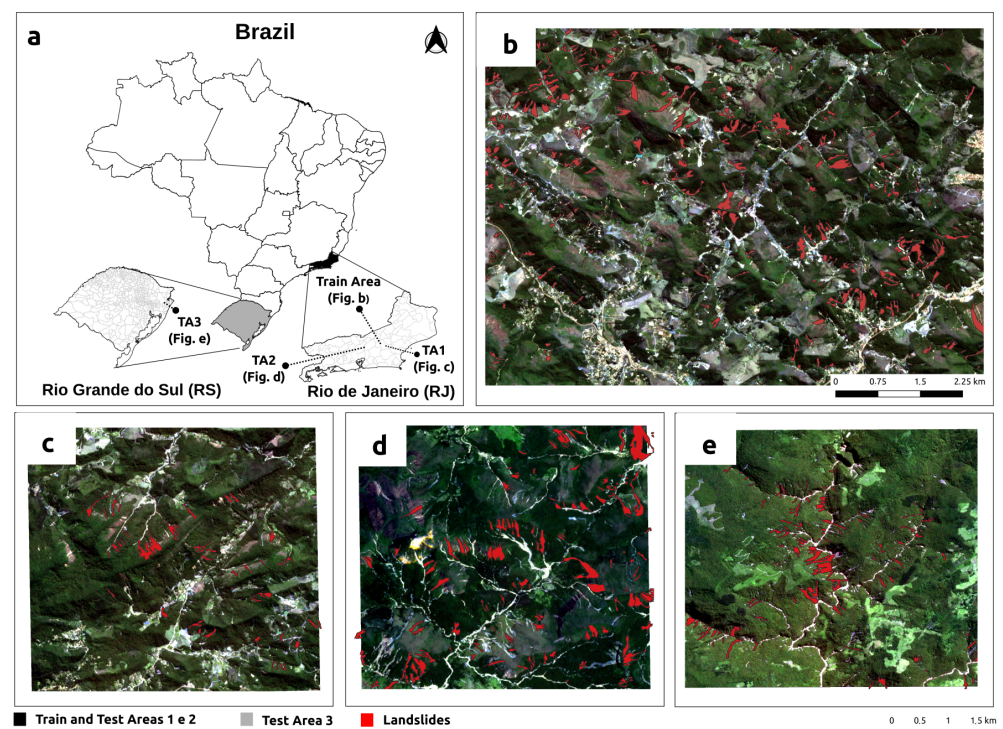

Figure 1. Location of the train and test areas used to train and evaluate the deep learning models. (**a**) Location of the train and test areas in Brazil. (**b**) Train Area. (**c**) Test Area 1 (TA1). (**d**) Test Area 2 (TA2). (**e**) Test Area 3 (TA3).

2.1. Nova Friburgo and Teresópolis

The mountainous region of Rio de Janeiro encompasses the municipalities of Nova Friburgo, Teresópolis, Petrópolis, Sumidouro, São José do Vale do Rio Preto, and Bom Jardim. In January 2011, an extreme rainfall event (140 mm/h) triggered at least 3500 translational landslides that killed more than 1500 people and disrupted all major city facilities in this mountainous region [11]. This event is considered the worst Brazilian natural disaster [70].

Nova Friburgo and Teresópolis are in the geomorphological unit of Serra dos Orgãos. The geological units have a WSW-ENE trend, and the elevation ranges between 1100 and 2000 m a.s.l. [71]. The geology consists mainly of igneous and metamorphic rocks such as

granites, diorites, gabbros, and gneisses [72] (Figure 2a). According to Köppen's climate classification scheme [73], the climate is subtropical highland (Cwb), with dry winters and mild summers. The annual mean precipitation is 1585.62 mm, with most of the rainfall in November, December, and January [74].

2.2. Rolante

The Rolante River Catchment has a drainage area of 828 km^2, with altitudes varying from 19 to 997 m a.s.l [75]. The area is inserted in the geomorphological unit of Serra Geral, with a predominance of basaltic rocks and sandstones (Figure 2b). The climate is characterized as very humid subtropical, with precipitation annual average between 1700 and 2000 mm. On 5 January 2017, an extreme precipitation event (272 mm in four hours) triggered at least 300 shallow landslide events in the area [75–78]. The flash flood caused by the material that moved from the slopes into the Mascara river (a tributary of the Rolante River) reached Rolante city.
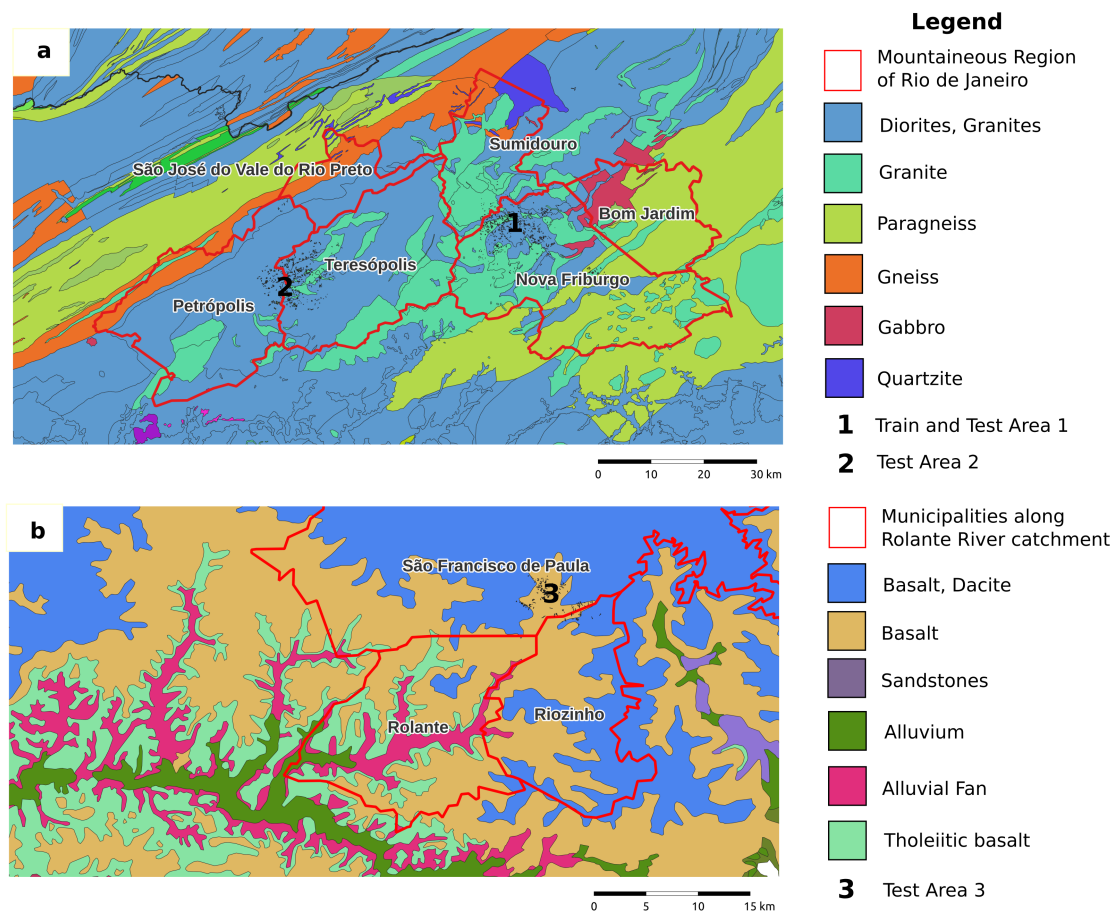

Figure 2. Simplified geological maps of the study areas. (**a**) Geological map of the mountainous region of Rio de Janeiro. (**b**) Geological map of the Rolante River area.

3. Methodology

The methodology applied in this study consists of four parts: pre-processing, training, evaluation, and post-processing (Figure 3). In the pre-processing step, the data were prepared to serve as the input to the U-Net models. Three different datasets were created to train the models. The sampling was done with regular grids in four different patch sizes: 32 × 32, 64 × 64, 128 × 128, 256 × 256. Augmentation consisted of random rotations, vertical and horizontal flips, and was used to keep the sample size the same among the different patch sizes. The training was done using the Tensorflow 2.0 Python Deep Learning Framework and used grid search to find the optimal hyperparameters. The

evaluation step used precision, recall, f1 score, and mean intersection over union (mIoU) to evaluate the accuracy of the models and their generalization capacity. The models were tested in three different areas with different scene complexities and locations. The post-processing step consisted of evaluating binary opening, closing, dilation, and erosion morphological operations.
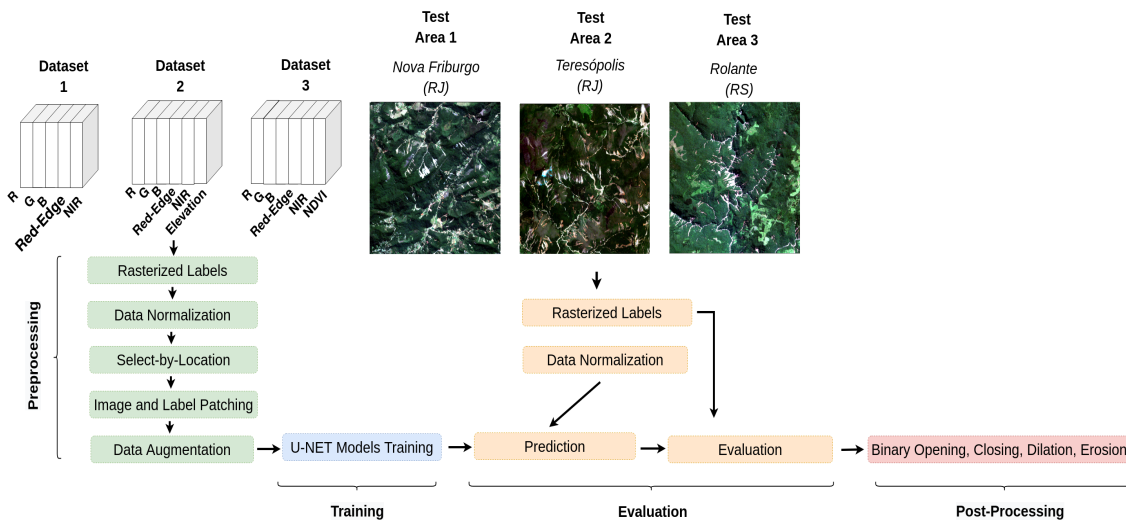

Figure 3. Workflow used to prepare the dataset and train, evaluate, and post-process the deep learning models and the results. "R", "G", "B", "Red-Edge", and "NDVI" represent the bands red, green, blue, red-edge, and the normalized vegetation index, respectively.

The data used in this study consist of the spectral information from the RapidEye satellite and topographical data from the Shuttle Radar Topography Mission (SRTM–[79]). RapidEye consists of a constellation of five identical satellites with high-resolution sensors with a 6.5 m nominal ground sampling distance at nadir. The orthorectified products are resampled and provided to users at a pixel size of 5 m. The data are acquired with a temporal resolution of 5 days in five spectral bands: blue (440–510 nm), green (520–590 nm), red (630–685 nm), red-edge (690–730 nm), near-infrared (760–850 nm) [80]. The SRTM acquired interferometric radar data with dual antennas and provided data with 1 arc-second (30 m) spatial resolution. The mission used single-pass interferometry radar to acquire two signals simultaneously by using two different radar antennas. Differences between the two signals permit the calculation of surface elevation [81].

This work used the RapidEye 3A product (orthorectified, radiometric, and geometric corrections) and was acquired from the Planet Explorer website [79]. The acquisition dates of the training and test images are in Table 1. The SRTM product was the 1 arc-second global (30 m).

Three datasets were generated to train and evaluate the deep learning models. All the datasets used the five RapidEye bands. However, dataset 1 used only those five bands, while in dataset 2, the elevation information was added as an extra channel in the image, and in dataset 3, the Normalized Difference Vegetation Index (NDVI) [82] was calculated (Equation (1)) using the red and the near-infrared (NIR) bands and added as an extra channel.

$$NDVI = \frac{NIR - RED}{NIR + RED} \tag{1}$$

The landslides were interpreted from the RapidEye and Google Earth Pro version 7.3 imagery and validated with [10,75] to minimize interpretation errors. Table 1 shows the number of landslide polygons interpreted in each scene. Later, the landslides were rasterized using the Rasterio Python library [83] to a binary mask, on which "1" represents

the landslides and "0" the background. The satellite images were normalized to convert all the pixel values into a 0–1 range interval. All the image pixel values were divided by 2^{16} (16 bits image). Data normalization helps in model convergence and is a common procedure in the machine learning field.

Table 1. Train and test images acquisition date, and the number of landslides present on each scene.

Images	Acquisition Date (RapidEye/SRTM)	Number of Landslides
Train Area (Nova Friburgo)	10 January 2011/23 September 2014	455
TA1 (Nova Friburgo)	10 January 2011/23 September 2014	42
TA2 (Teresópolis)	20 January 2011/23 September 2014	117
TA3 (Rolante)	13 March 2017/23 September 2014	110

The data were sampled with regular grids in four sizes: 32×32, 64×64, 128×128, 256×256 pixels. Patching the data in different sizes is an important step to address the differences in the shapes and sizes of the landslides [61]. Moreover, since the patch sizes are directly correlated with the balance between the positive (landslides) and the negative (background) classes, training the models with different sizes is crucial to determine the optimal size for the best model performance in the study areas. A select-by-location operation was used to select only the polygons intersecting landslides. This process ensures that all sampled images will have a small portion of a landslide scar, reducing class imbalance.

Data augmentation allows the use of the annotated data more efficiently during the training phase [67,84]. In this work, because the data were sampled in different patch sizes, the smaller patch sizes have more samples than the larger ones. Hence, comparing the models trained with varying patch sizes may not be fair as the different sample sizes may affect the training of the deep learning models [4]. Thus, to keep the same sample size for all the models, augmentation processes of random rotations and vertical and horizontal flips were performed in the sampled data with patch sizes of 64×64, 128×128, and 256×256 pixels.

3.1. U-Net

U-Net [85] is a fully convolutional network developed for the segmentation of biomedical images. This type of architecture does not use fully connected layers in their structure; instead, they have an encoder–decoder architecture with just convolutional layers (Figure 4). The encoder path is responsible for classifying the pixels without taking the spatial location into account, while the decoder path uses up-convolutions and concatenation to recover the spatial location of the classified pixels and return a mask with the same dimensions of the input image.

The convolutional blocks on the encoder path have two 3×3 convolutional layers, activated with the Rectified Linear Unit (ReLU) function, and followed by a max-pooling operation that reduces the spatial dimension by 2. The dropout layer was used with a 0.5 probability after each max-pooling to randomly deactivate some of the layers of the network as a method to reduce the overfitting.

The convolutional layers are responsible for creating feature maps of the input image to allow the model to predict the landslide. During the training step, the 3×3 kernels present in these layers are calibrated to find specific features of the landslides. The nonlinear activation function ReLU was calculated according to Equation (2). The use of ReLU increased the degrees of freedom of the computed function, which allows the model to learn nonlinear patterns present in the data [86]. The max-pooling layers with 2×2 kernels

translate around the image, obtaining only the highest values and reducing the image dimensions by half. This operation is essential to reduce the computation cost and to preserve the values with the highest relevance. Dropout [87] layers are commonly used in the training phase to reduce the complexity of the model and, consequently, the overfitting with random deactivation of the layers with a p probability. In the architecture used in this study, the dropouts were implemented after each max-pooling layer in the encoder path.

On the decoder path, 2 × 2 up-sampling operations increase the data's spatial dimension to concatenate feature maps with the same dimension from the encoder path. Then, the concatenated data serve as input for two convolutional layers before another up-sampling operation. At the last layer, a sigmoid function converts the output into a binary mask. The 2 × 2 kernels of the transposed convolutions learn how to increase the dimensions of the feature maps during the training step and increase the size of the feature maps by 2. The sigmoid function (Equation (3)) converts the values to the 0–1 range at the last layer.

$$\text{ReLU} = max(0, x) \tag{2}$$
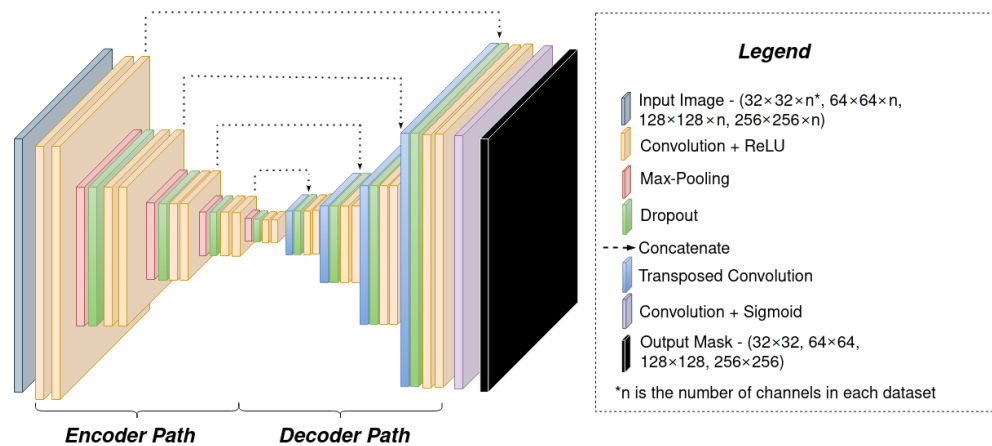
$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

Figure 4. U-Net network architecture.

The models were trained for 200 epochs with a dynamic learning rate of 0.001 that reduces by 0.1 in a loss function plateau. Binary Cross Entropy and Adam function were used as the loss and optimization function, respectively. The models were trained with four different batch sizes (16, 32, 64, 128 samples). The model's weights were saved when the validation loss function decreased to reduce the overfitting. The models were trained on Keras [88] and Tensorflow 2.0 [89] Python libraries. Moreover, 30% of each dataset was used as validation data. The training was held in a NVIDIA™ GeForce RTX 2060 GPU (8 GB memory, NVIDIA, Santa Clara, CA, USA).

3.2. Validation Metrics

The model's performance was evaluated over two test areas by using the f1 score, recall, precision, and mean intersection over union (mIoU) metrics. These metrics are based on true positives (TP), false positives (FP), and false negatives (FN) [61,90,91]. TP are pixels correctly classified as landslides. FP represents the pixels incorrectly classified as landslides, and FN the pixels incorrectly classified as the background. The models that were trained with DEM and NDVI as an additional channel were evaluated on test areas with an additional DEM and NDVI channel. Precision (Equation (4)) defines how accurate the model is by evaluating how many of the classified areas are landslides. The metric is useful for evaluating the cost of false positives. Recall (Equation (5)) calculates how many of the actual positives are true positives. This metric is suitable to evaluate the cost

associated with false negatives. The f1 score (Equation (6)) combines precision and recall to measure if there is a balance between true positives and false negatives. Mean intersection over union (Equation (7)), also known as the Jaccard Index, computes the overlapping of areas between the ground truth (*A*) and the model prediction (*B*) divided by the union of these areas. Then, the values are averaged for each class. *A* value of 1 (one) represents perfect overlapping, while 0 (zero) represents no overlap.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{4}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{5}$$

$$\text{f1-Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

$$\text{mIoU} = \frac{A \cap B}{A \cup B} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives} + \text{False Negatives}} \tag{7}$$

3.3. Post-Processing

In this study, post-processing morphological operations were used to optimize the results. Binary opening, closing, erosion, and dilation operators were evaluated individually and combined to find the greater improvement (Figure 5). The binary opening helps in removing minor errors that do not represent landslide candidates. Meanwhile, closing, which consists of a dilation followed by erosion, fills the holes inside predicted landslides [67]. Erosion is a mathematical morphological operation that erodes the boundaries of the foreground to shrink the landslide candidates and enlarge the background. Dilation opening helps in removing small noises (i.e., "salt") in the landslide prediction and connects small dark cracks. This tends to open background gaps between the landslides [92]. Several parameters were tested to find the optimal configuration for the post-processing operations. The best structuring element was a 3×3 square and the interaction was done until the results did not change anymore.
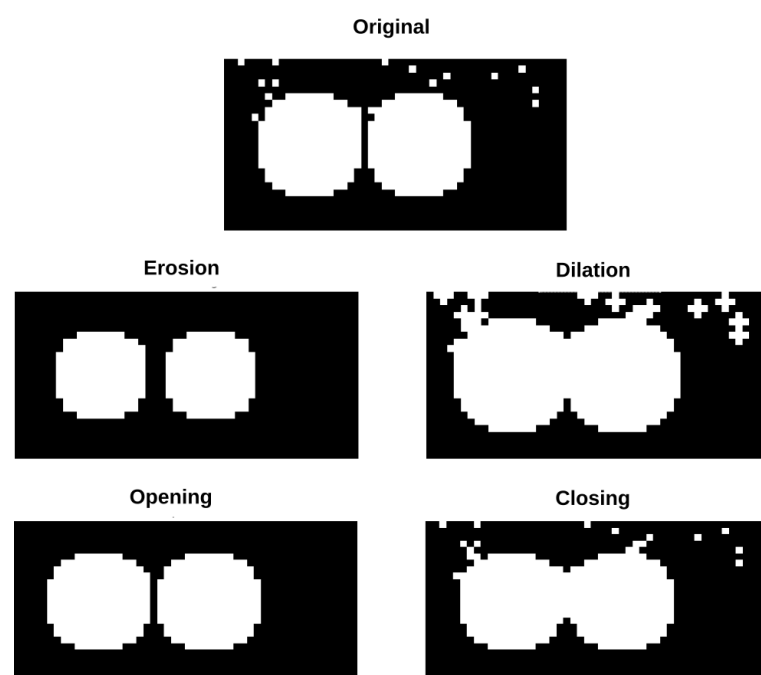

Figure 5. Morphological operations used to post-process the segmentation results.

4. Results and Discussion

The models were trained with four different patch sizes. In general, models trained with all patch sizes learned the feature maps to detect the landslides. The result shows (Figure 6) that the models trained with 32 × 32 and 64 × 64 pixels achieved the best f1 scores in TA2 (0.53) and TA3 (0.60). In contrast, models trained with 128 × 128 pixels patches achieved the best f1 score results in TA1 (0.52). Since TA1 is located close to the training area, the results show that the models trained with larger patches became better in detecting landslides similar to the training images. This occurs because the patches with greater dimensions facilitate the understanding of the global scene context. Consequently, the deep learning model specialized in detecting landslides with similar spectral and morphological characteristics to the training images. On the other hand, the models trained with the smaller tiles learn the local context of the landslide better. Therefore, they make excessive predictions (low precision), reducing the f1 and mIoU in TA1. However, they achieve better results in TA2 and TA3.

Ghorbanzadeh et al. [61] and Soares et al. [93] evaluated samples with different patch sizes to address the difference in landslide shapes. Ghorbanzadeh et al. [61] conclude that the patch sizes affected the results in a non-systematic way. Meanwhile, Soares et al. [93] observed that models trained with larger patches achieved higher precision and lower recall. Similar results were observed by Prakash et al. [68], where the authors trained the models with 224 × 224 pixels and obtained results with bias towards high precision and lower recall. In this study, the results show that the models tend to achieve better precision and lower recall rates with larger patch sizes. Moreover, comparing the results achieved, it is possible to see that this pattern is more evident in TA1 than the other areas. Thus, once models trained with larger patch sizes become highly specialized in detecting the shape and spectral characteristics of the training area, they tend to achieve better precision in those areas and have worse results in the regions that differ from the training regions.
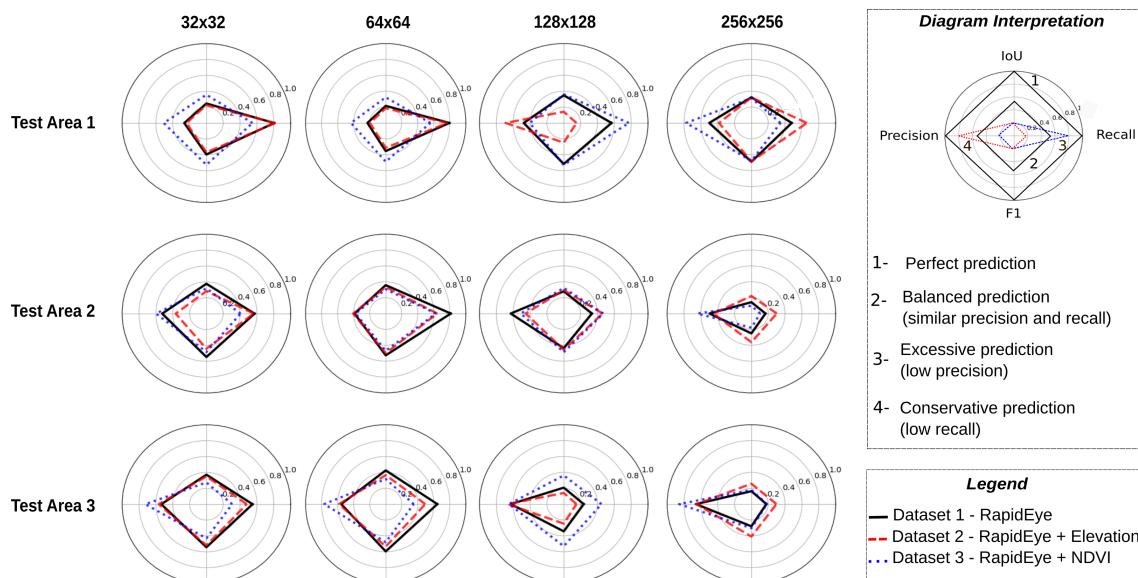

Figure 6. F1 score, precision, recall, and mIoU results of the best segmentation models trained with different patch sizes.

Dataset 3 achieved more balanced results than the other datasets, while dataset 1 achieved better f1 and mIoU scores in most models. The higher balance of dataset 3 may be related to the extra NDVI band. NDVI is a band normalization computation with values that range from 0 to 1 and are comparable even in different images. Consequently, it provides information that facilitates the model generalization. Furthermore, dataset 1 has a lower dimensionality (five bands); therefore, according to the Hughes Phenomena [94], it needs less data to train the model than the other datasets with higher dimensionality

(six bands). The topographical data do not improve the results, which are in accordance with the results obtained by Sameen and Pradhan [66]. This may be related to the greater dimensionality of dataset 3 and the SRTM spatial resolution (30 m).

Spectral indices, such as NDVI, are commonly used in remote sensing to help in the interpretation of the spectral signatures of various objects [95]. The correct selection of features based on these indices is crucial in improving traditional machine learning algorithms [96–98]. However, there is a tradeoff between the number of samples and the dimensionality of the data [99]. The extra bands with spectral indices may not improve the algorithm's performance if the dataset is not large enough to overcome the Hughes Phenomena. Moreover, the deep learning convolution operations may learn to calculate the NDVI in the training process from the spectral bands, and the extra band will be redundant. To the best of our knowledge, only the study of Ghorbanzadeh et al. [100] evaluated the impact of using spectral bands and topographic factors (slope, aspect, plan curvature, elevation). However, in this study, the NDVI was used as the basis for landslide detection and was not evaluated; the model architecture used is a classification network that predicts in a pixel-wise manner. For fully convolutional networks, such as U-Net, still, no study evaluates each band's impact on the model performance.

Evaluating the histogram of the best model results in each test area (Figure 7) and it is possible to see that in TA1, the model prediction achieves higher true and false positive rates than the other test areas. Meanwhile, in the other test areas, false negative results were higher. This pattern shows that despite the models' generalization capacity, the areas with different environmental and spectral characteristics from the training area made the model more restrictive. Therefore, fewer landslides are predicted correctly, and the number of false negatives is greater. Prakash et al. [68] observed a similar pattern, where the models trained with different study areas were biased towards high false negatives. Thus, the false positives of TA1 may represent landslides missing in the ground truth inventory, which does not directly represent a poor result.
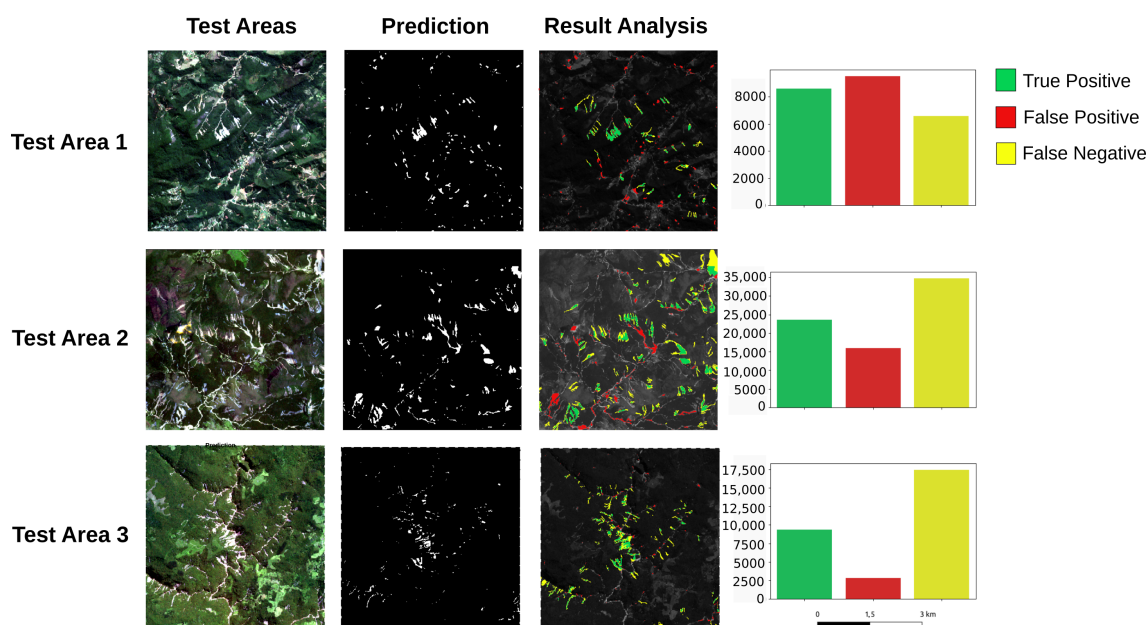
Figure 7. Segmentation results of each test area and result histograms showing the number of pixels representing true positives, false positives, and true negatives. These results correspond to the prediction of the model trained with 128 × 128 pixel tiles and dataset 3 for TA1; the model trained with 32 × 32 tile pixels and dataset 1 for TA2, and the model trained with 64 × 64 pixels and dataset 1 TA3.

The complexity of the scene is also an essential factor in evaluating the generalization capacity of the models. In previous studies [61,67,68], the test areas are usually vegetated

areas around the landslide scars. Models trained and tested with these scenes may not be efficient to detect landslides in urban areas due to the higher complexity of the scene and may not be feasible for applications in disaster scenarios. In Qi et al. [98], the authors noticed that the deep learning models had difficulties distinguishing roads and buildings from landslides. In this study, the test areas were chosen to represent areas with different characteristics and complexity. As shown in the histogram of Figure 8, the scores of the models evaluated on TA1 and TA2, which are close to Nova Friburgo and Teresópolis, were reduced by false positives caused by roads and the roofs of the houses. The errors occur in areas with similar spectral responses to the landslides. Since the spatial resolution of the RapidEye images used in this study is 5 m, the model cannot differentiate the shape of the landslides from rivers with increased bedload, areas with bare soil, roads, and roofs. Consequently, the models made these mistakes in all areas. It was expected that the models trained with the DEM layer would overcome the misclassification of the drainage and urban areas since these areas usually have different terrain morphological attributes such as slope and aspect. Probably, these errors occurred due to the coarse resolution of the available DEM (30 m), which cannot clearly detach objects and generalizes the terrain. In Ghorbanzadeh et al. [61], the authors used a 5 m DEM and observed that the DEM helped in differentiating the human settlement areas.
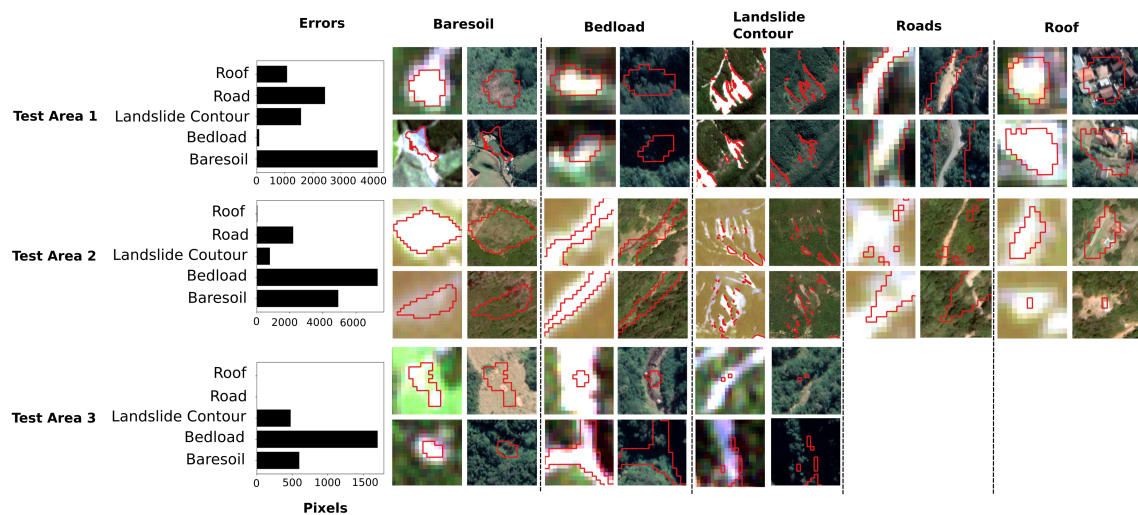

Figure 8. Comparison of the most frequent errors made by the deep learning models during landslide segmentation and histograms showing the number of pixels for each error category. Images with lower resolution (**left**) are from the RapidEye satellite, and images with higher resolution (**right**) are from Google Earth.

The post-processing operations were efficient in improving the precision of all test areas. The precision values improved from 0.56 to 0.64 in TA1, 0.57 to 0.65 in TA2, and 0.64 to 0.81 in TA3 (Figure 9). The results of all operations are given in the Supplementary Material Table S1. These results show that the post-processing techniques are efficient in removing the model's systematic errors and are efficient for improving the segmentation precision results.
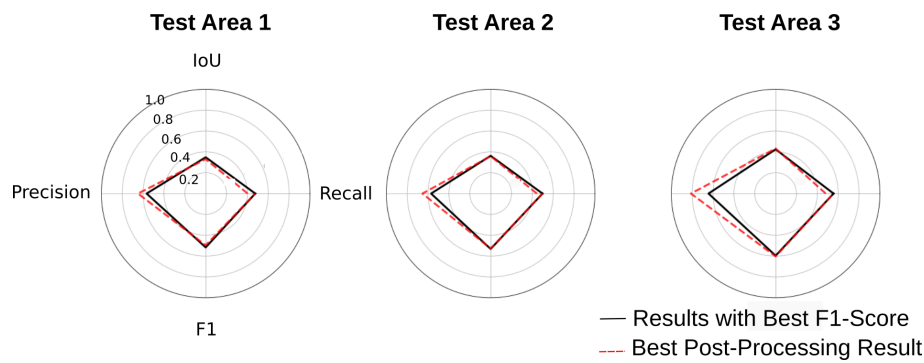
Figure 9. F1 score, recall, precision, and mIoU of the best segmentation results of each test area after the post-processing operations. The data used for the post-processing operations correspond to the prediction of the model trained with 128 × 128 pixel tiles and dataset 3 for TA1; the model trained with 32 × 32 pixel tiles and dataset 1 for TA2, and the model trained with 64 × 64 pixel tiles and dataset 1 for TA3.

The morphological operations were evaluated individually and in combination; Table 2 shows the best three combinations' average results for each test area. TA2 and TA3 achieved the best results with the same operations (dilation; closing/dilation; erosion/opening/closing), while in TA1, the operations that yielded the best results were opening; erosion/dilation; and dilation/erosion/opening. Such similarity in the post-processing of TA2 and TA3 and the difference with TA1 may be related to the environmental differences in the training area and the prediction pattern of the model. TA3 achieved better precision results in comparison with the other test areas. This difference seems to be related to each area's landslide characteristics and the model results. Therefore, post-processing operations cannot be generalized and different operations should be tested to find the optimal solution.

Table 2. Best post-processing operations and results for each test area. The best results were calculated by summing all the result values. Values in bold represent the best results before the post-processing operations.

Area	Operation	Recall	Precision	F1-Score	mIoU
TA1	**-**	**0.57**	**0.47**	**0.52**	**0.35**
TA1	Opening Erosion + Dilation Dilation + Erosion + Opening	0.48	0.56	0.52	0.35
TA2	**-**	**0.50**	**0.57**	**0.53**	**0.36**
TA2	Dilation Closing + Dilation Erosion + Opening + Closing	0.44	0.65	0.53	0.36
TA3	**-**	**0.56**	**0.64**	**0.60**	**0.42**
TA3	Dilation Closing + Dilation Erosion + Opening + Closing	0.48	0.81	0.60	0.43

5. Conclusions

This study evaluated the generalization capacity of deep learning models and post-processing techniques. The results show that the patch size highly affects the prediction accuracy in areas that are different from the training zone. The larger patch improved the test area results that were close to the training area because larger patches favor a global comprehension of the scene. Consequently, the model becomes specialized in detecting landslides similar to the ones used for the training. On the other hand, the models trained with the smaller patches achieved better results in TA2 and TA3 in locations different from the training zone. This is because the models trained with smaller patches understand the

local context better; they can predict the landslides in a more satisfactory way in different locations. Nevertheless, they also tend to be more restrictive and make more false negative errors. The complexity of the scene is directly correlated with the performance of the models. Therefore, comparing results obtained from different authors, and from different data acquisition methods, such as lidar and Remote Piloted Aircrafts (RPA), may not be reasonable since each training and test area has its own characteristics and complexities. In this way, to better evaluate the machine and deep learning models, a future effort should be made towards an open dataset to evaluate landslide deep learning models. Such open datasets are standard in other computer vision studies such as ImageNet [101], MNIST [102], EuroSat [103] UC Merced Land Use Dataset [104], AID dataset [105], and Brazilian Coffee Scene [106]. Post-processing the results is an efficient step to improve the precision of the segmentation results. The TA3 results improved by 0.17 after combining binary erosion, opening, and closing. The best method to post-process the results will depend on the landslides' characteristics and the model results. Therefore, one should test different combinations and parameters in a semi-supervised way to find the optimal solution. The use of spectral indexes seems to help in balancing the precision and recall of the models and improving model generalization. Since spectral indexes have comparable ranges that facilitate model convergence, the calculation of these indexes is important for predicting landslides in areas with different characteristics from the training areas. Future work should evaluate whether the use of these indexes also facilitates landslide detection in images from different sensors and resolutions.

References

1. Metternicht, G.; Hurni, L.; Gogu, R. Remote sensing of landslides: An analysis of the potential contribution to geo-spatial systems for hazard assessment in mountainous environments. *Remote Sens. Environ.* **2005**, *98*, 284–303. [CrossRef]
2. Froude, M.J.; Petley, D.N. Global fatal landslide occurrence from 2004 to 2016. *Nat. Hazards Earth Syst. Sci.* **2018**, *18*, 2161–2181. [CrossRef]
3. Schuster, R.L.; Highland, L.M. The Third Hans Cloos Lecture. Urban landslides: Socioeconomic impacts and overview of mitigative strategies. *Bull. Eng. Geol. Environ.* **2007**, *66*, 1–27. [CrossRef]
4. Yi, Y.; Zhang, Z.; Zhang, W.; Zhang, C.; Li, W.; Zhao, T. Semantic segmentation of urban buildings from VHR remote sensing imagery using a deep convolutional neural network. *Remote Sens.* **2019**, *11*, 1774. [CrossRef]
5. Hong, H.; Chen, W.; Xu, C.; Youssef, A.M.; Pradhan, B.; Tien Bui, D. Rainfall-induced landslide susceptibility assessment at the Chongren area (China) using frequency ratio, certainty factor, and index of entropy. *Geocarto Int.* **2017**, *32*, 139–154. [CrossRef]

6. Alexander, D.E. A brief survey of GIS in mass-movement studies, with reflections on theory and methods. *Geomorphology* **2008**, *94*, 261–267. [CrossRef]

7. Zhong, C.; Liu, Y.; Gao, P.; Chen, W.; Li, H.; Hou, Y.; Nuremanguli, T.; Ma, H. Landslide mapping with remote sensing: Challenges and opportunities. *Int. J. Remote Sens.* **2019**, *41*, 1555–1581. [CrossRef]

8. Tominaga, L.K.; Santoro, J.; Amaral, R. *Desastres Naturais*; Instituto Geológico: São Paulo, Brazil, 2009.

9. CRED. EM-DAT: The International Emergency Disasters Database. 2022. Available online: https://www.emdat.be/ (accessed on 1 May 2022).

10. Coelho-Netto, A.L.; de Souza Avelar, A.; Lacerda, W.A. Landslides and disasters in southeastern and southern Brazil. *Dev. Earth Surf. Process.* **2009**, *13*, 223–243.

11. Netto, A.L.C.; Sato, A.M.; de Souza Avelar, A.; Vianna, L.G.G.; Araújo, I.S.; Ferreira, D.L.; Lima, P.H.; Silva, A.P.A.; Silva, R.P. January 2011: The extreme landslide disaster in Brazil. In *Landslide Science and Practice*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 377–384.

12. Vieira, B.C.; Gramani, M.F. Serra do Mar: The most "tormented" relief in Brazil. In *Landscapes and Landforms of Brazil*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 285–297.

13. Mondini, A.C.; Santangelo, M.; Rocchetti, M.; Rossetto, E.; Manconi, A.; Monserrat, O. Sentinel-1 SAR Amplitude Imagery for Rapid Landslide Detection. *Remote Sens.* **2019**, *11*, 760. [CrossRef]

14. Guzzetti, F.; Mondini, A.C.; Cardinali, M.; Fiorucci, F.; Santangelo, M.; Chang, K.T. Landslide inventory maps: New tools for an old problem. *Earth-Sci. Rev.* **2012**, *112*, 42–66. [CrossRef]

15. Catani, F.; Tofani, V.; Lagomarsino, D. Spatial patterns of landslide dimension: A tool for magnitude mapping. *Geomorphology* **2016**, *273*, 361–373. [CrossRef]

16. Mavroulis, S.; Diakakis, M.; Kranis, H.; Vassilakis, E.; Kapetanidis, V.; Spingos, I.; Kaviris, G.; Skourtsos, E.; Voulgaris, N.; Lekkas, E. Inventory of Historical and Recent Earthquake-Triggered Landslides and Assessment of Related Susceptibility by GIS-Based Analytic Hierarchy Process: The Case of Cephalonia (Ionian Islands, Western Greece). *Appl. Sci.* **2022**, *12*, 2895. [CrossRef]

17. Shao, X.; Ma, S.; Xu, C.; Shen, L.; Lu, Y. Inventory, distribution and geometric characteristics of landslides in Baoshan City, Yunnan Province, China. *Sustainability* **2020**, *12*, 2433. [CrossRef]

18. Ardizzone, F.; Basile, G.; Cardinali, M.; Casagli, N.; Del Conte, S.; Del Ventisette, C.; Fiorucci, F.; Garfagnoli, F.; Gigli, G.; Guzzetti, F.; et al. Landslide inventory map for the Briga and the Giampilieri catchments, NE Sicily, Italy. *J. Maps* **2012**, *8*, 176–180. [CrossRef]

19. Conforti, M.; Muto, F.; Rago, V.; Critelli, S. Landslide inventory map of north-eastern Calabria (South Italy). *J. Maps* **2014**, *10*, 90–102. [CrossRef]

20. Dias, H.C.; Hölbling, D.W.; Grohmann, C.H. Landslide inventory mapping in Brazil: Status and challenges. In Proceedings of the XIII International Symposium on Landslides, virtual, 22–26 February 2021.

21. Marcelino, E.V.; Formagio, A.R. Análise comparativa entre métodos heurísticos de mapeamento de áreas suscetíveis a escorregamento. *Simpósio Bras. Desastr. Nat.* **2004**, *1*, 392–407.

22. Schulz, W.H. Landslide susceptibility revealed by LIDAR imagery and historical records, Seattle, Washington. *Eng. Geol.* **2007**, *89*, 67–87. [CrossRef]

23. Kawabata, D.; Bandibas, J. Landslide susceptibility mapping using geological data, a DEM from ASTER images and an Artificial Neural Network (ANN). *Geomorphology* **2009**, *113*, 97–109. [CrossRef]

24. Chen, H.; Zeng, Z. Deformation prediction of landslide based on improved back-propagation neural network. *Cogn. Comput.* **2013**, *5*, 56–62. [CrossRef]

25. Du, J.C.; Teng, H.C. 3D laser scanning and GPS technology for landslide earthwork volume estimation. *Autom. Constr.* **2007**, *16*, 657–663. [CrossRef]

26. Lingua, A.; Piatti, D.; Rinaudo, F. Remote monitoring of a landslide using an integration of GB-INSAR and LIDAR techniques. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2008**, *37*, 133–139.

27. Baldo, M.; Bicocchi, C.; Chiocchini, U.; Giordan, D.; Lollino, G. LIDAR monitoring of mass wasting processes: The Radicofani landslide, Province of Siena, Central Italy. *Geomorphology* **2009**, *105*, 193–201. [CrossRef]

28. Jaboyedoff, M.; Oppikofer, T.; Abellán, A.; Derron, M.H.; Loye, A.; Metzger, R.; Pedrazzini, A. Use of LIDAR in landslide investigations: A review. *Nat. Hazards* **2012**, *61*, 5–28. [CrossRef]

29. Brideau, M.A.; McDougall, S.; Stead, D.; Evans, S.G.; Couture, R.; Turner, K. Three-dimensional distinct element modelling and dynamic runout analysis of a landslide in gneissic rock, British Columbia, Canada. *Bull. Eng. Geol. Environ.* **2012**, *71*, 467–486. [CrossRef]

30. Ghuffar, S.; Székely, B.; Roncat, A.; Pfeifer, N. Landslide displacement monitoring using 3D range flow on airborne and terrestrial LiDAR data. *Remote Sens.* **2013**, *5*, 2720–2745. [CrossRef]

31. Casagli, N.; Cigna, F.; Bianchini, S.; Hölbling, D.; Füreder, P.; Righini, G.; Del Conte, S.; Friedl, B.; Schneiderbauer, S.; Iasio, C.; et al. Landslide mapping and monitoring by using radar and optical remote sensing: Examples from the EC-FP7 project SAFER. *Remote Sens. Appl. Soc. Environ.* **2016**, *4*, 92–108. [CrossRef]

32. Hsiao, K.; Liu, J.; Yu, M.; Tseng, Y. Change detection of landslide terrains using ground-based LiDAR data. In Proceedings of the XXth ISPRS Congress, Istanbul, Turkey, Commission VII, WG, Istanbul, Turkey, 12–13 July 2004; Volume 7, p. 5.

33. McKean, J.; Roering, J. Objective landslide detection and surface morphology mapping using high-resolution airborne laser altimetry. *Geomorphology* **2004**, *57*, 331–351. [CrossRef]

34. Glenn, N.F.; Streutker, D.R.; Chadwick, D.J.; Thackray, G.D.; Dorsch, S.J. Analysis of LiDAR-derived topographic information for characterizing and differentiating landslide morphology and activity. *Geomorphology* **2006**, *73*, 131–148. [CrossRef]

35. Dewitte, O.; Jasselette, J.C.; Cornet, Y.; Van Den Eeckhaut, M.; Collignon, A.; Poesen, J.; Demoulin, A. Tracking landslide displacements by multi-temporal DTMs: A combined aerial stereophotogrammetric and LIDAR approach in western Belgium. *Eng. Geol.* **2008**, *99*, 11–22. [CrossRef]

36. Kasai, M.; Ikeda, M.; Asahina, T.; Fujisawa, K. LiDAR-derived DEM evaluation of deep-seated landslides in a steep and rocky region of Japan. *Geomorphology* **2009**, *113*, 57–69. [CrossRef]

37. Liu, J.K.; Chang, K.T.; Rau, J.Y.; Hsu, W.C.; Liao, Z.Y.; Lau, C.C.; Shih, T.Y. The geomorphometry of rainfall-induced landslides in taiwan obtained by airborne lidar and digital photography. In *Geoscience and Remote Sensing*; In-Tech, Inc.: Garching Bei München, Germany, 2009.

38. Burns, W.J.; Coe, J.A.; Kaya, B.S.; Ma, L. Analysis of elevation changes detected from multi-temporal LiDAR surveys in forested landslide terrain in western Oregon. *Environ. Eng. Geosci.* **2010**, *16*, 315–341. [CrossRef]

39. Ventura, G.; Vilardo, G.; Terranova, C.; Sessa, E.B. Tracking and evolution of complex active landslides by multi-temporal airborne LiDAR data: The Montaguto landslide (Southern Italy). *Remote Sens. Environ.* **2011**, *115*, 3237–3248. [CrossRef]

40. Van Westen, C.J.; Castellanos, E.; Kuriakose, S.L. Spatial data for landslide susceptibility, hazard, and vulnerability assessment: An overview. *Eng. Geol.* **2008**, *102*, 112–131. [CrossRef]

41. Korup, O.; Stolle, A. Landslide prediction from machine learning. *Geol. Today* **2014**, *30*, 26–33. [CrossRef]

42. Steger, S.; Brenning, A.; Bell, R.; Glade, T. The influence of systematically incomplete shallow landslide inventories on statistical susceptibility models and suggestions for improvements. *Landslides* **2017**, *14*, 1767–1781. [CrossRef]

43. Nilsen, T.H. *Preliminary Photointerpretation Map of Landslide and Other Surficial Deposits of the Concord 15-Minute Quadrangle and the Oakland West, Richmond, and Part of the San Quentin 7 1/2-Minute Quadrangles, Contra Costa and Alameda Counties, California*; Technical Report; US Geological Survey: Reston, VA, USA, 1973.

44. Guzzetti, F.; Carrara, A.; Cardinali, M.; Reichenbach, P. Landslide hazard evaluation: A review of current techniques and their application in a multi-scale study, Central Italy. *Geomorphology* **1999**, *31*, 181–216. [CrossRef]

45. Van Den Eeckhaut, M.; Poesen, J.; Verstraeten, G.; Vanacker, V.; Moeyersons, J.; Nyssen, J.; Van Beek, L. The effectiveness of hillshade maps and expert knowledge in mapping old deep-seated landslides. *Geomorphology* **2005**, *67*, 351–363. [CrossRef]

46. Booth, A.M.; Roering, J.J.; Perron, J.T. Automated landslide mapping using spectral analysis and high-resolution topographic data: Puget Sound lowlands, Washington, and Portland Hills, Oregon. *Geomorphology* **2009**, *109*, 132–147. [CrossRef]

47. Burns, W.J.; Madin, I. Protocol for Inventory Mapping of Landslide Deposits from Light Detection and Ranging (LiDAR) Imagery. 2009. Available online: https://www.oregongeology.org/pubs/dds/slido/sp-42_onscreen.pdf (accessed on 1 May 2022).

48. Roering, J.J.; Mackey, B.H.; Marshall, J.A.; Sweeney, K.E.; Deligne, N.I.; Booth, A.M.; Handwerger, A.L.; Cerovski-Darriau, C. 'You are HERE': Connecting the dots with airborne lidar for geomorphic fieldwork. *Geomorphology* **2013**, *200*, 172–183. [CrossRef]

49. Scaioni, M.; Longoni, L.; Melillo, V.; Papini, M. Remote sensing for landslide investigations: An overview of recent achievements and perspectives. *Remote Sens.* **2014**, *6*, 9600–9652. [CrossRef]

50. Van Den Eeckhaut, M.; Hervás, J.; Jaedicke, C.; Malet, J.P.; Montanarella, L.; Nadim, F. Statistical modelling of Europe-wide landslide susceptibility using limited landslide inventory data. *Landslides* **2012**, *9*, 357–369. [CrossRef]

51. Knevels, R.; Petschko, H.; Leopold, P.; Brenning, A. Geographic object-based image analysis for automated landslide detection using open source GIS software. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 551. [CrossRef]

52. Li, Y.; Chen, W. Landslide susceptibility evaluation using hybrid integration of evidential belief function and machine learning techniques. *Water* **2020**, *12*, 113. [CrossRef]

53. Wang, Y.; Fang, Z.; Wang, M.; Peng, L.; Hong, H. Comparative study of landslide susceptibility mapping with different recurrent neural networks. *Comput. Geosci.* **2020**, *138*, 104445. [CrossRef]

54. Xu, C. Preparation of earthquake-triggered landslide inventory maps using remote sensing and GIS technologies: Principles and case studies. *Geosci. Front.* **2015**, *6*, 825–836. [CrossRef]

55. Yu, B.; Chen, F.; Xu, C. Landslide detection based on contour-based deep learning framework in case of national scale of Nepal in 2015. *Comput. Geosci.* **2020**, *135*, 104388. [CrossRef]

56. Blaschke, T. Object based image analysis for remote sensing. *ISPRS J. Photogramm. Remote Sens.* **2010**, *65*, 2–16. [CrossRef]

57. Hossain, M.D.; Chen, D. Segmentation for Object-Based Image Analysis (OBIA): A review of algorithms and challenges from remote sensing perspective. *ISPRS J. Photogramm. Remote Sens.* **2019**, *150*, 115–134. [CrossRef]

58. Stumpf, A.; Kerle, N. Object-oriented mapping of landslides using Random Forests. *Remote Sens. Environ.* **2011**, *115*, 2564–2577. [CrossRef]

59. Blaschke, T.; Hay, G.J.; Kelly, M.; Lang, S.; Hofmann, P.; Addink, E.; Feitosa, R.Q.; Van der Meer, F.; Van der Werff, H.; Van Coillie, F.; et al. Geographic object-based image analysis–towards a new paradigm. *ISPRS J. Photogramm. Remote Sens.* **2014**, *87*, 180–191. [CrossRef]

60. Prakash, N.; Manconi, A.; Loew, S. Mapping landslides on EO data: Performance of deep learning models vs. traditional machine learning models. *Remote Sens.* **2020**, *12*, 346. [CrossRef]

61. Ghorbanzadeh, O.; Blaschke, T.; Gholamnia, K.; Meena, S.R.; Tiede, D.; Aryal, J. Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection. *Remote Sens.* **2019**, *11*, 196. [CrossRef]

62. Peng, D.; Zhang, Y.; Guan, H. End-to-End Change Detection for High Resolution Satellite Images Using Improved UNet++. *Remote Sens.* **2019**, *11*, 1382. [CrossRef]

63. 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. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [CrossRef]

64. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.

65. Radovic, M.; Adarkwa, O.; Wang, Q. Object recognition in aerial images using convolutional neural networks. *J. Imaging* **2017**, *3*, 21. [CrossRef]

66. Sameen, M.I.; Pradhan, B. Landslide Detection Using Residual Networks and the Fusion of Spectral and Topographic Information. *IEEE Access* **2019**, *7*, 114363–114373. [CrossRef]

67. Yi, Y.; Zhang, W. A new deep-learning-based approach for earthquake-triggered landslide detection from single-temporal RapidEye satellite imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 6166–6176. [CrossRef]

68. Prakash, N.; Manconi, A.; Loew, S. A new strategy to map landslides with a generalized convolutional neural network. *Sci. Rep.* **2021**, *11*, 1–15. [CrossRef]

69. Pradhan, B.; Seeni, M.I.; Nampak, H. Integration of LiDAR and QuickBird data for automatic landslide detection using object-based analysis and random forests. In *Laser Scanning Applications in Landslide Assessment*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 69–81.

70. Avelar, A.S.; Netto, A.L.C.; Lacerda, W.A.; Becker, L.B.; Mendonça, M.B. Mechanisms of the recent catastrophic landslides in the mountainous range of Rio de Janeiro, Brazil. In *Landslide Science and Practice*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 265–270.

71. Dantas, M.E. *Geomorfologia do Estado do Rio de Janeiro*; CPRM. Estudo Geoambiental do Estado do Rio de Janeiro: Brasília, Brazil, 2001.

72. Tupinambá, M.; Heilbron, M.; Duarte, B.P.; de Almeida, J.C.H.; Valladares, C.S.; Pacheco, B.T.; dos Santos Salomão, M.; Conceição, F.R.; da Silva, L.G.E.; de Almeida, C.G.; et al. *Mapa Geológico Folha Nova Friburgo SF-23-Z-B-II*; Technical Report; CPRM—Serviço Geológico do Brasil: Rio de Janeirom, Brazil, 2012.

73. Köppen, W. *Das Geographische System der Klimate*; Gerbrüder Borntraeger: Stuttgart, Germany, 1936; Volume 1, Das geographische System der Klimate; pp. 1–44.

74. Sobral, B.S.; Oliveira-Júnior, J.F.; Gois, G.; de Bodas Terassi, P.M.; Muniz-Júnior, J.G.R. Variabilidade espaço-temporal e interanual da chuva no estado do Rio de Janeiro. *Rev. Bras. Climatol.* **2018**, *22*. [CrossRef]

75. Uehara, T.D.T.; Correa, S.P.L.P.; Quevedo, R.P.; Körting, T.S.; Dutra, L.V.; Rennó, C.D. Classification algorithms comparison for landslide scars. *GEOINFO* **2019**, *20*, 158.

76. Gameiro, S.; Quevedo, R.P.; Oliveira, G.; Ruiz, L.; Guasselli, L. Análise e correlação de atributos morfométricos e sua influência nos movimentos de massa ocorridos na Bacia do Rio Rolante, RS. In Proceedings of the Anais do XIX Simpósio Brasileiro de Sensoriamento Remoto, Santos, Brazil, 14–17 April 2019; pp. 2880–2883.

77. Quevedo, R.P.; Oliveira, G.; Gameiro, S.; Ruiz, L.; Guasselli, L. Modelagem de áreas suscetíveis a movimentos de massa com redes neurais artificiais. In Proceedings of the Anais do XIX Simpósio Brasileiro de Sensoriamento Remoto, Santos, Brazil, 14–17 April 2019; Volume 19, pp. 2910–2913.

78. Quevedo, R.P.; Guasselli, L.A.; Oliveira, G.G.; Ruiz, L.F.C. Modelagem de áreas suscetíveis a movimentos de massa: Avaliação comparativa de técnicas de amostragem, aprendizado de máquina e modelos digitais de elevação. *Geociências* **2020**, *38*, 781–795. [CrossRef]

79. Farr, T.G.; Rosen, P.A.; Caro, E.; Crippen, R.; Duren, R.; Hensley, S.; Kobrick, M.; Paller, M.; Rodriguez, E.; Roth, L.; et al. The Shuttle Radar Topography Mission. *Rev. Geophys.* **2007**, *45*, RG2004. [CrossRef]

80. RapidEye, A. Satellite imagery product specifications. In *Satellite Imagery Product Specifications: Version*; BlackBridge: Lethbridge, AB, Canada, 2011.

81. Planet Team. Planet Application Program Interface: In Space for Life on Earth. 2017. Available online: https://api.planet.com (accessed on 1 May 2020).

82. Tucker, C.J. Red and photographic infrared linear combinations for monitoring vegetation. *Remote Sens. Environ.* **1979**, *8*, 127–150. [CrossRef]

83. Rasterio: Geospatial Raster I/O for Python Programmers. 2013. Available online: https://github.com/rasterio/rasterio (accessed on 1 May 2022).

84. Dixit, M.; Kwitt, R.; Niethammer, M.; Vasconcelos, N. Aga: Attribute-guided augmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7455–7463.

85. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.

86. Quinn, J.; McEachen, J.; Fullan, M.; Gardner, M.; Drummy, M. *Dive into Deep Learning: Tools for Engagement*; Corwin Press: Thousand Oaks, CA, USA, 2019.

87. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

88. Keras. 2015. Available online: https://keras.io/ (accessed on 1 May 2022).

89. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: tensorflow.org (accessed on 1 May 2022).

90. Ghorbanzadeh, O.; Tiede, D.; Dabiri, Z.; Sudmanns, M.; Lang, S. Dwelling extraction in refugee camps using cnn–first experiences and lessons learnt. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**. [CrossRef]

91. Guirado, E.; Tabik, S.; Alcaraz-Segura, D.; Cabello, J.; Herrera, F. Deep-learning convolutional neural networks for scattered shrub detection with google earth imagery. *arXiv* **2017**, arXiv:1706.00917.

92. Van der Walt, S.; Schönberger, J.L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J.D.; Yager, N.; Gouillart, E.; Yu, T. Scikit-image: Image processing in Python. *PeerJ* **2014**, *2*, e453. [CrossRef]

93. Soares, L.P.; Dias, H.C.; Grohmann, C.H. Landslide segmentation with u-net: Evaluating different sampling methods and patch sizes. *arXiv* **2020**, arXiv:2007.06672.

94. Hughes, G. On the mean accuracy of statistical pattern recognizers. *IEEE Trans. Inf. Theory* **1968**, *14*, 55–63. [CrossRef]

95. Verstraete, M.M.; Pinty, B. Designing optimal spectral indexes for remote sensing applications. *IEEE Trans. Geosci. Remote Sens.* **1996**, *34*, 1254–1265. [CrossRef]

96. Mondini, A.; Guzzetti, F.; Reichenbach, P.; Rossi, M.; Cardinali, M.; Ardizzone, F. Semi-automatic recognition and mapping of rainfall induced shallow landslides using optical satellite images. *Remote Sens. Environ.* **2011**, *115*, 1743–1757. [CrossRef]

97. Lu, P.; Qin, Y.; Li, Z.; Mondini, A.C.; Casagli, N. Landslide mapping from multi-sensor data through improved change detection-based Markov random field. *Remote Sens. Environ.* **2019**, *231*, 111235. [CrossRef]

98. Qi, W.; Wei, M.; Yang, W.; Xu, C.; Ma, C. Automatic mapping of landslides by the ResU-net. *Remote Sens.* **2020**, *12*, 2487. [CrossRef]

99. Altman, N.; Krzywinski, M. The curse (s) of dimensionality. *Nat. Methods* **2018**, *15*, 399–400. [CrossRef] [PubMed]

100. Ghorbanzadeh, O.; Meena, S.R.; Abadi, H.S.S.; Piralilou, S.T.; Zhiyong, L.; Blaschke, T. Landslide Mapping Using Two Main Deep-Learning Convolution Neural Network Streams Combined by the Dempster–Shafer Model. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *14*, 452–463. [CrossRef]

101. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]

102. Bottou, L.; Cortes, C.; Denker, J.S.; Drucker, H.; Guyon, I.; Jackel, L.D.; LeCun, Y.; Muller, U.A.; Sackinger, E.; Simard, P.; et al. Comparison of classifier methods: A case study in handwritten digit recognition. In Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3-Conference C: Signal Processing (Cat. No. 94CH3440-5), Jerusalem, Israel, 9–13 October 1994; Volume 2, pp. 77–82.

103. Helber, P.; Bischke, B.; Dengel, A.; Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 2217–2226. [CrossRef]

104. Yang, Y.; Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, San Jose, CA, USA, 2–5 November 2010; pp. 270–279.

105. Xia, G.S.; Hu, J.; Hu, F.; Shi, B.; Bai, X.; Zhong, Y.; Zhang, L.; Lu, X. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]

106. Penatti, O.A.; Nogueira, K.; Dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 7–12 June 2015; pp. 44–51.

7.3 *Landslide detection in the Himalayas using machine learning algorithms and U-Net*

Sansar Raj Meena · **Lucas Pedrosa Soares** · **Carlos H. Grohmann** ·
Cees van Westen · **Kushanav Bhuyan** · **Ramesh P. Singh** · **Mario Floris** ·
Filippo Catani

Landslide detection in the Himalayas using machine learning algorithms and U-Net

Abstract Event-based landslide inventories are essential sources to broaden our understanding of the causal relationship between triggering events and the occurring landslides. Moreover, detailed inventories are crucial for the succeeding phases of landslide risk studies like susceptibility and hazard assessment. The openly available inventories differ in the quality and completeness levels. Event-based landslide inventories are created based on manual interpretation, and there can be significant differences in the mapping preferences among interpreters. To address this issue, we used two different datasets to analyze the potential of U-Net and machine learning approaches for automated landslide detection in the Himalayas. Dataset-1 is composed of five optical bands from the RapidEye satellite imagery. Dataset-2 is composed of the RapidEye optical data, and ALOS-PALSAR derived topographical data. We used a small dataset consisting of 239 samples acquired from several training zones and one testing zone to evaluate our models' performance using the fully convolutional U-Net model, Support Vector Machines (SVM), K-Nearest Neighbor, and the Random Forest (RF). We created thirty-two different maps to evaluate and understand the implications of different sample patch sizes and their effect on the accuracy of landslide detection in the study area. The results were then compared against the manually interpreted inventory compiled using fieldwork and visual interpretation of the RapidEye satellite image. We used accuracy assessment metrics such as F1-score, Precision, Recall, and Mathews Correlation Coefficient (MCC). In the context of the Nepali Himalayas, employing RapidEye images and machine learning models, a viable patch size was investigated. The U-Net model trained with 128×128 pixel patch size yields the best MCC results (76.59%) with the dataset-1. The added information from the digital elevation model benefited the overall detection of landslides. However, it does not improve the model's overall accuracy but helps differentiate human settlement areas and river sand bars. In this study, the U-Net achieved slightly better results than other machine learning approaches. Although it can depend on architecture of the U-Net model and the complexity of the geographical features in the imagery, the U-Net model is still preliminary in the domain of landslide detection. There is very little literature available related to the use of U-Net for landslide detection. This study is one of the first efforts of using U-Net for landslide detection in the Himalayas. Nevertheless, U-Net has the potential to improve further automated landslide detection in the future for varied topographical and geomorphological scenes.

Introduction

Loss of property and human life due to earthquake-triggered landslides is significantly high and is expected to increase due to climate change (Froude and Petley 2016; Gariano and Guzzetti 2016). About 47,000 earthquake-induced landslide casualties were reported from 2004 to 2010 (Petley 2012). Earthquake-induced landslides (EQIL) have direct and indirect long-term socioeconomic and environmental effects (Fan et al. 2018). The direct and indirect effects of landslides, for example, through the formation and breakout of landslide dams, are a significant natural hazard in the mountain regions of the Himalayas (Dhital 2015). Studies show unprecedented loss to both human lives and the economy in the Himalayan regions due to landslides, contributing up to 30% of the world's total landslide-related damage value (Dahal and Hasegawa 2008; Haigh and Rawat 2011). In Northern India, for example, during the recent 2021 Uttarakhand landslides, 24 people were killed by landslides and around 150 were missing (Meena et al. 2021a). A large number of people are affected in the Himalayan regions by small and large-scale landslides, especially during the monsoon seasons (Khanal and Watanabe 2005; Thapa and Dhital 2000; Upreti and Dhital 1996). Although landslides often occur in remote areas, the resulting catastrophic flash floods from landslide dam outbreak cause extensive damage to settlements, hydroelectric projects, and agriculture fields in the downstream areas (Meena and Tavakkoli Piralilou 2019).

To better analyze the frequency and distribution of landslides, there is a growing demand for event-based inventories that can be used to determine the probability of landslide occurrence in space and time as a basis for hazard and risk assessment. There is still insufficient information on landslide occurrences for many areas to make reliable hazard maps (Reichenbach et al. 2018). Landslide susceptibility and hazard modeling require accurate and complete landslide inventory datasets. This inventory dataset is usually used for training hazard models to find potential landslide-prone areas (Guzzetti et al. 2012).

The accuracy and completeness of landslide inventory datasets are essential for making spatial predictions for future events (Hakan and Luigi 2020). The mapping of event-based landslide inventories in remote and mountainous areas makes remote sensing data the primary source of information for mapping these events (Chen et al. 2018).

In terms of detecting landslide boundaries with remote sensing images, classification methods like pixel-based, feature-based, and object-based techniques can be employed (Lu et al. 2020; Su et al.

2020). While pixel-based methods only extract features by classifying each pixel, they do not take the spatial-context into account. However, feature-based methods (like gray level co-occurrence matrix and principal component analysis) (Whitworth et al. 2002) and object-based image analysis (OBIA) explicitly leverage the spatial information from satellite images (Bacha et al. 2020; Hölbling et al. 2012; Martha et al. 2010). During the last decade, deep-learning models and other machine learning models, particularly Convolutional Neural Networks (CNNs), have been applied successfully in a broad range of image segmentation and object detection purposes (Ding et al. 2016; Ghorbanzadeh et al. 2020; Jin et al. 2019; Liu et al. 2019; Shi et al. 2020).

The use of CNN models has yielded promising results for classification of aerial images (Bui et al. 2019; Ghorbanzadeh et al. 2021, 2020; Meena et al. 2021b; Yu et al. 2017). Numerous studies using CNN have been conducted for landslide detection (see Table 1). Many authors used CNN models for automated landslide detection in mountainous regions using multi-temporal high-resolution remote sensing data, mono-temporal medium-resolution image data (Chen et al. 2018), where others optimized their models and compared with existing baseline models such as Fully Convolutional Networks (FCNs) (Lei et al. 2019). Hyperspectral data for landslide detection was first investigated by Ye et al. (2019). In recent studies, different topographical factors like elevation and its derivates like slope, aspect, and curvature combined with remote sensing data for landslide detection were explored to improve landslide detection (Sameen and Pradhan 2019; Liu et al. 2020b; Prakash et al. 2020).

Deep learning models usually require large amount of training data to detect objects efficiently. However, since landslide inventories are generated for small areas using manual interpretation and fieldwork, such inventories commonly have just a few samples and present a limitation for the training of deep learning models (Chen et al. 2020; Liu et al. 2020a; Qi et al. 2020) Therefore, in this study, the main objective was to evaluate and compare the performance of the machine and deep learning models trained with a small dataset composed of only 239 landslide polygons (55 polygons for training and 184 polygons for testing purposes). The fully convolutional U-Net deep learning model and other machine learning models were trained with data from a 5-m RapidEye optical satellite imagery and resampled 12.5-m ALOS PALSAR digital elevation data for landslide detection.

Study area

The study area is located in Rasuwa district Nepal, which is situated in higher Himalayas and is one of the highly affected regions after the 2015 Gorkha earthquake (see Fig. 1). Most of the study area falls in the Langtang national park and there are several hydropower plants projects along the Trishuli River. After the 2015 Gorkha earthquake, a series of landslides triggered by the earthquake caused damage to hydro powerplants, agricultural land, and human settlements. On 15 April 2015, during the Gorkha earthquake, more than 80 people were killed due to EQILs and flood events near the Mailung village hydropower plant camps. Several attempts have been made by local authorities and foreign institutes to study impact of landslide on human settlements and hydro powerplants in the region. However, in many inaccessible hilly areas, field visit

was not feasible hence remote sensing tools can help supplement the field visits. The study area is highly affected by monsoonal rains and every year several deep-seated landslides get reactivated such as the one near Ramche village.

Data used and methodology

Datasets

The landslides were visually interpreted as polygons from RapidEye imagery acquired on 04 November 2016 (Planet Labs Inc.) and field observations. The data has 5 m spatial resolution in five spectral bands: blue (440–510 nm), green (520–590 nm), red (630–685 nm), red-edge (690–730 nm), and near-infrared (760–850 nm) (RapidEye 2011).

A total of 239 landslide polygons were mapped in the entire study area, 55 in the training zones and 184 in the test zone (the training zones are yellow and testing zones are red in color in Fig. 1c). For training the model, 117 sampling points were manually selected along the centerline of the landslide polygons present in the training zone. Other 57 points were selected outside the landslide polygons to represent non-landslide samples (see Fig. 1c). Therefore, a total of 174 sampling points were used to train the models. Those points were used as the centroid to generate the training patches of four sizes: 16×16, 32×32, 64×64, and 128×128 pixels (Fig. 2).

Two datasets were created to train the models. Dataset-1 consists of the five spectral bands (RGB, red-edge, NIR) from the RapidEye satellite. The Dataset-2 consists of the same five bands and two extra topographical bands (elevation and slope). The elevation and slope data were acquired from a digital elevation model (DEM), resampled to 5-m spatial resolution, derived from Phased Array type L-band Synthetic Aperture Radar (PALSAR) of the Advanced Land Observing Satellite (ALOS).

All the models used the same training data to compare the results from the models properly. The deep learning algorithms were trained using the Python libraries TensorFlow 2.0 and the machine learning using Scikit-Learn.

Classifiers

U-Net model

U-Net (Ronneberger et al. 2015) is a state-of-art deep learning model used for semantic segmentation tasks. This model has an encoder-decoder architecture similar to the letter "U" (Fig. 3). The encoder path is composed of blocks of two 3×3 convolutional layers followed by a 2×2 max-pooling layer. The convolutional layers are 3×3 moving windows that translate around the image, calculating a dot product that can be summarized by Eq. 1 (Zhang et al. 2018):

$$O^l = \sigma(O^{l-1} * W^l + b^l) \tag{1}$$

where O^{l-1} refers to the output of the (l-1)th layer, W^l represents the weights and b^l represents the bias. σ indicates the non-linear activation function. The rectified linear unit (ReLU) was used as the activation function in this research. ReLU is commonly used as the

Table 1 Overview of some recently published studies on the automated mapping of landslides using deep learning approaches sorted according to the use of topographical features

Main objective	Algorithms used	Topographical feature used	Accuracy evaluation methods	Study
Optimization of (FCN-PP) for landslide inventory mapping	ELSE, RLSE, CDMRF, CDFFCM, CNN, FCN, U-Net, FCN-PP	N/A	Precision, recall, F1-score, OE, accuracy	(Lei et al. 2019)
Automatic mapping of landslides by the ResU-Net	U-Net, ResU-Net	N/A	Precision, recall, F1-score	(Qi et al. 2020)
Landslide recognition by deep convolutional neural network and change detection	CNN, CDCNN	DEM	Precision, recall, F1-score	(Shi et al. 2020)
To detect landslides of Nepal in the year 2015, at a national scale	ResNet-101 CNN	DEM	Precision, recall, F1-score	(Yu et al. 2020)
Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks	VGG16, ResNet	DEM	Precision, recall, F1-score, Accuracy	(Ji et al. 2020)
An automated approach for landslide detection	DCNN	Slope	CE, DP, QP	(Chen et al. 2018)
Landslide detection using Hyperspectral remote sensing data and comparison of conventional methods with DLWC	DLWC, SVM, SID, SAM	Slope	Overall accuracy, Kappa coefficient, accuracy	(Ye et al. 2019)
Landslide detection using UAV–derived VHR imagery and topographical factors	CNN	Slope	PPV, TPR, F1-score, OPR, UPR, mIOU	(Ghorbanzadeh et al. 2019a)
Post-earthquake landslide extraction using the U-Net model	U-Net, U-Net + ResNet	DSM, slope, slope aspect	Precision, recall, F1-score, mIOU	(Liu et al. 2020b)
Comparison of different sample patch sizes for landslide detection using deep learning and machine learning	CNN, SVM, D-CNN, RF, ANN	Plan curvature, slope, slope aspect	Precision, recall, F1-score, mIOU	(Ghorbanzadeh et al. 2019b)
Landslide detection using residual networks	ResNet, CNN	Altitude, slope, slope aspect, total curvature	Training accuracy, validation accuracy, F1-score, mIOU	(Sameen and Pradhan 2019)
Comparison of pixel-based, object-based, and deep learning methods for landslide detection	RF, ANN, LR, U-Net + ResNet	Hill shade, slope, aspect. Terrain roughness, curvature, valley depth, TWI	Accuracy, F1-score, MCC, POD, POFD	(Prakash et al. 2020)

FCN-PP fully convolutional network within pyramid pooling, *ELSE* employed edge-based level set evolution, *RLSE* region-based level set evolution, *CDMRF* change detection-based on Markov random field, *CDFFCM* change detection-based fast FCM, *CNN* convolutional neural network, *FCN* fully convolutional network, *ResU-Net*, residual and U-Net, *VGG16* very deep convolutional networks for large-scale image recognition, *UAV* unmanned aerial vehicle, *D-CNN* deep convolution neural network, *DLWC*, deep learning with constrains, *SVM* support vector machine, *SID* spectral information divergence, *SAM* spectral angle match, *RF* random forest, *ANN* artificial neural networks, *LR* logistic regression, *ResNet* residual networks
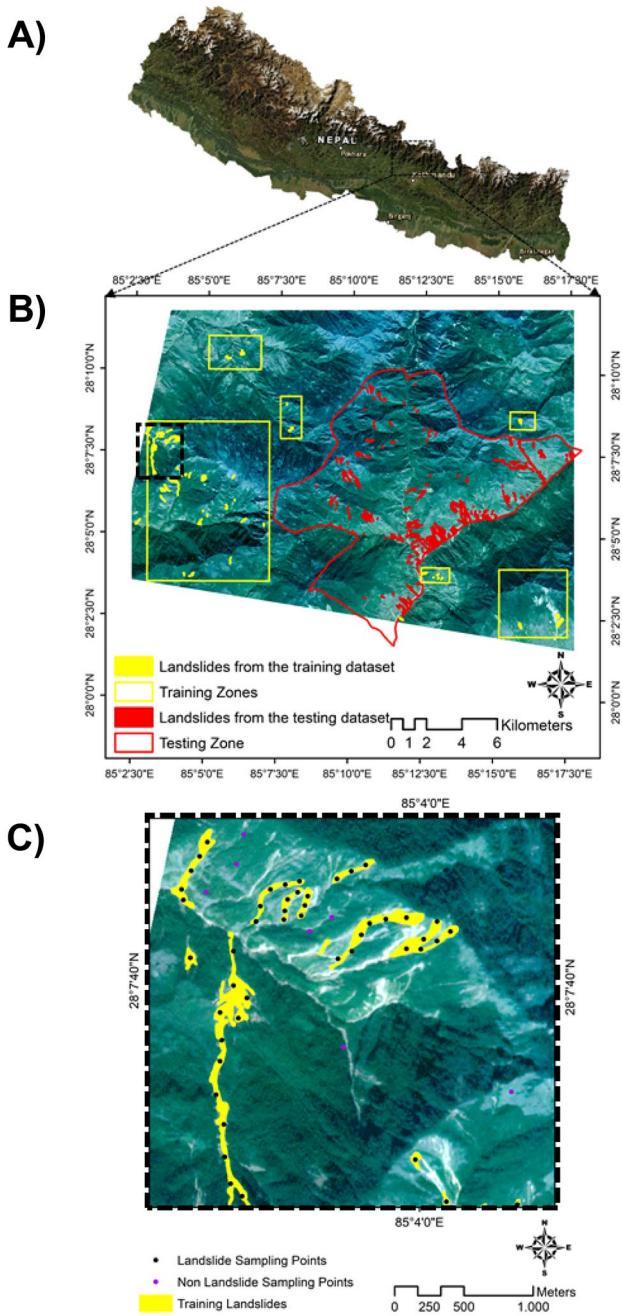
A)

B)

C)

Fig. 1 **A** Location of the study area in Nepal, **B** landslide training and testing zones in the study area, and **C** sampling points along the center line of the landslide polygons (black) and non-landslide class (purple)

activation function because it is more efficient than other functions and reduces the gradient vanishing problem during the training step (Wang et al. 2019). The function returns 0 when the input is negative and the same input value if it is positive. The max-pooling layers keep only the maximum values from the feature maps generated from the convolution operation. Thus, after a max-pooling

operation, the spatial dimension of the feature map is reduced to half of the input size.

The decoder path recovers the spatial location by using up-convolutions and concatenations from the encoder path (Ronneberger et al. 2015). The up-convolution layers increase the dimensions of the feature maps. The layers' output is concatenated with the feature map from the symmetrical position in the encoder path. In the last layer, a sigmoid function was used to output the class predictions in a 0–1 probability range. A threshold of 0.5 was used to determine the positive (> 0.5) and the negative (< 0.5) classes after the prediction.

Several papers describe and explain the U-Net structure and how convolutional neural networks are trained (Ghorbanzadeh et al. 2019a, b; Prakash et al. 2020; Wang et al. 2019). In this study, we use a fully convolutional neural network that is capable of calculating per-pixel probability of comprising a landslide. Unlike previous work conducted by Ghorbanzadeh et al. (2019a, b) where they used a classical convolutional neural network to generate patchwise landslide classification, the neural network used in our study is more efficient for landslide segmentation problems as the result is a binary output with the same size as the input image (Prakash et al. 2020, 2021; Qi et al. 2020). The network hyperparameter tuning process considered different number of filters (8, 16, 32), learning rates (0.01, 0.001, 0.0001), and batch sizes (8, 16, 32). The learning rate value was reduced by a factor of 0.1 when the validation loss function reaches a plateau for more than twenty epochs. The models were saved only when the validation loss function decreased as an attempt to avoid overfitting.

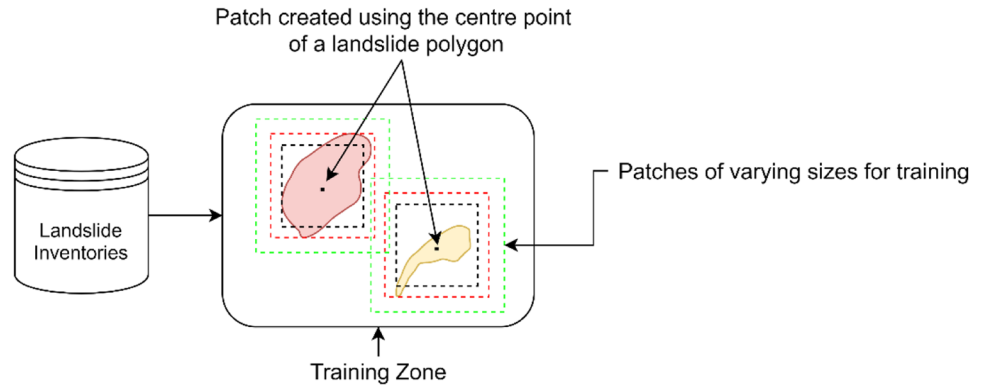Support vector machine (SVM) model

SVM is a machine learning method that uses kernel functions to map the dataset into a higher dimension to determine a hyperplane that separates the training data feature spaces (Cortes and Vapnik 1995). The margining of the hyperplanes, also known as support vectors, is maximized to be the closest to the training features. This method gained popularity for landslide mapping due to accurate results, even with small datasets and unknown statistical distributions (Moosavi et al. 2014; Mountrakis et al. 2011; Pawłuszek and Borkowski 2016).

The classification result is affected by the kernel function (e.g., linear, sigmoid, polynomial, radial basis). Thus, various kernel functions were evaluated to find the best classifier.

K-nearest neighbors algorithm (KNN) model

K-nearest neighbors is a machine learning algorithm that uses the training data to find the feature space's K-closest neighbors. The algorithm outputs a class probability that reflects the uncertainty with which a given individual item can be assigned to any given class (Marjanovic et al. 2009). In this study, the distance between the feature space points was calculated using the Euclidean distance method. An optimal K value was determined by testing K in a 1–10 range.

Fig. 2 Conceptualization of generating the patches for training the models

Patch created using the centre point of a landslide polygon

Patches of varying sizes for training

Landslide Inventories

Training Zone

Random forests (RF) model

Random forest is an ensemble method widely used for landslide detection (Chen et al. 2014). The method is based on multiple decision trees. Each tree is slightly different since they are trained with the training dataset's random subsets. The technique is less prone to overfitting because each tree's output class is weighted based on

a majority voting technique where the class with the most votes becomes the model's prediction.

Multiple input patches

Different patch sizes may affect the model accuracy because landslides have different shapes and sizes, which may not be
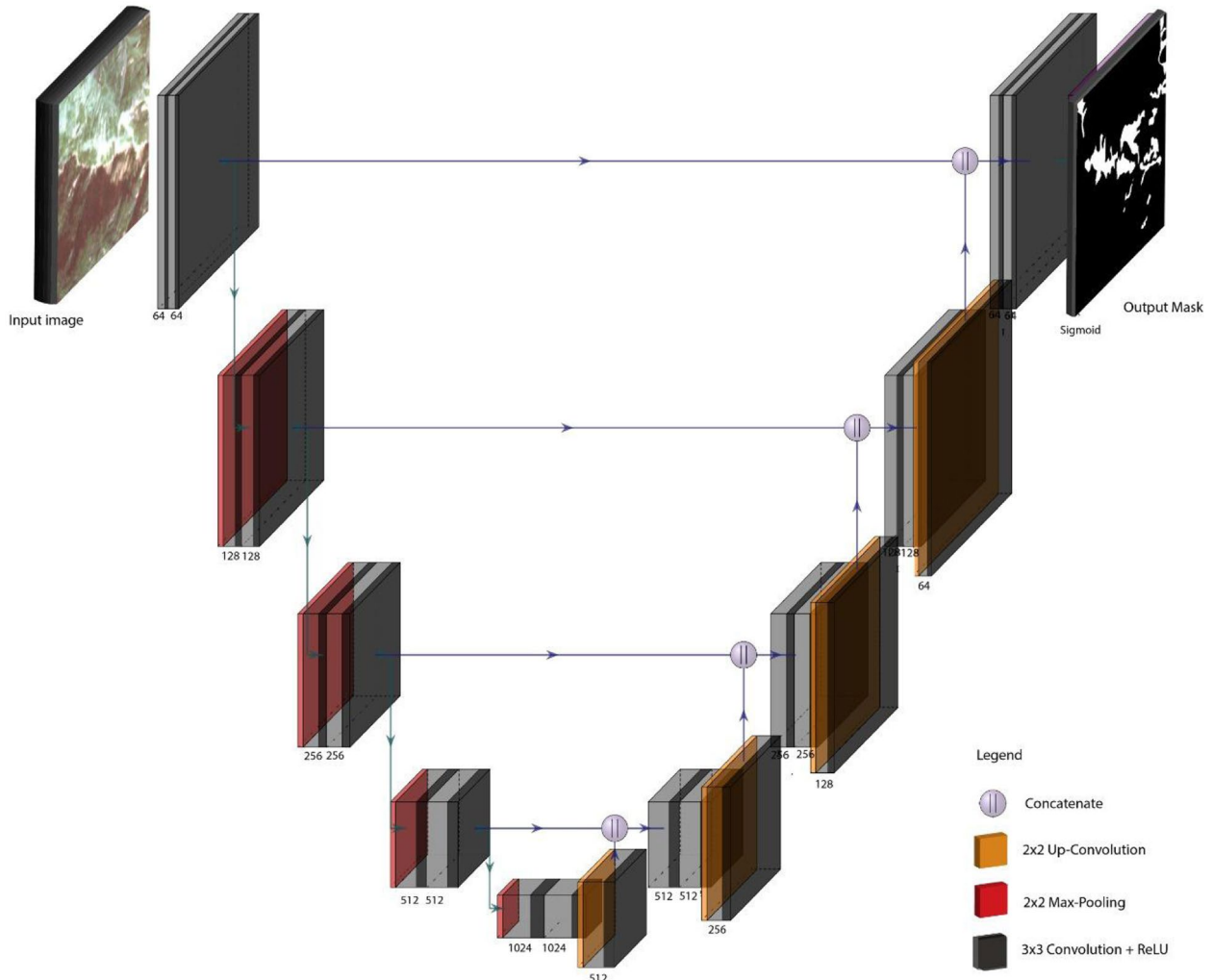

Fig. 3 The architecture of the U-Net model. The numbers below the convolution represent the number of filters used to train the model

well-represented depending on the patch size. Moreover, since the negative class is usually more frequent than the positive class in remote sensing imagery, larger patches may negatively influence the model because they can increase the imbalance between the positive and negative class (Ghorbanzadeh et al. 2019a, b).

In this work, the patches used to train the models were constituted by a multiple of 16 pixels since this is a condition to effectively train the U-Net model. The models were trained with 16×16, 32×32, 64×64, and 128×128 pixel patches to compare and evaluate how the different patch sizes affect the accuracy of the model. The models were also trained with 256×256 pixels patches. However, since the results were inferior compared to the other patch sizes, only the results achieved with the mentioned patch sizes were considered in the "Results" section.

Results

The machine and deep learning models were trained using only 174 samples to evaluate and compare the performance of the algorithms using small datasets. In total, sixteen result maps were generated for each dataset (dataset-1 and dataset-2). The result maps (Figs. 4a, b; 5a, b; 6a, b; 7a, b) are named based on the algorithm, the patch size, and the dataset used to train the algorithm. Therefore, the map U-Net_16_5 and U-Net_16_7 (Fig. 4a and b) correspond to the U-Net deep learning algorithm trained with the 16×16 patch size using the dataset with five optical bands (dataset-1) and seven bands (dataset-2), respectively. The best results were achieved by U-Net models with a learning rate of 0.001; SVM models trained with a polynomial kernel function and a scalable gamma parameter (γ); KNN models trained with nine neighbors; and RF models with 200 trees and depth 8.

Figure 8 portrays the differences in the areas of the landslides detected with the different machine learning models with respect to the influence of the topographical information from dataset-2. As seen in Fig. 8b, the total area in most of the models is relatively higher in dataset-2 than dataset-1 when compared against the manually interpreted ground truth area. This difference is because of the detection of false positives as an influence from the slope and elevation in dataset-2. Although there are improvements in the built-up area and river sand bars, the model gets confused and generates false positives in forests and agricultural areas.

The models were evaluated based on precision, recall, F1-score, and Matthews Correlation Coefficient (MCC) metrics, which are calculated using the value of true positive (TP), false positives (FP), and false negatives (FN) (Fig. 9). Precision (Eq. 2) calculates the proportion of pixels correctly classified as landslides. Recall (Eq. 3) value represents the number of pixels that was correctly classified as landslides from the total pixels representing landslides.

$$\text{Precision} = \frac{TP}{TP + FP} \times 100 \qquad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN} \times 100 \qquad (3)$$

F1-score (Eq. 4) is a harmonic mean between precision and recall; therefore, the highest values of F1-score correspond to

models with better performance. Landslide datasets usually have an unbalance between the positive (landslides) and negative (background) classes. Thus, the MCC (Eq. 5) metric is better for comparing imbalanced datasets (Baldi et al. 2000).

$$F1score = \frac{2 \times \text{precison} \times \text{recall}}{(\text{precision} + \text{recall})} \times 100 \qquad (4)$$

$$MCC = \frac{TP \times TN - FPxPN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \times 100 \qquad (5)$$

The results show that among the models trained with dataset-1, the U-Net 128,5 model achieved the highest MCC (71.06) and F1-score (71.12). Nevertheless, compared with the other algorithms, the MCC results are just 0.63, 1.59, and 2.65 higher than the SVM, KNN, and RF algorithms (Table 2). SVM 1285 achieved the highest precision (80.28), while U-Net 16,5 had the highest recall (83.94).

The U-Net also had better performance in dataset-2 (Table 3). However, in dataset-1, the model trained with 128×128 patch size achieved the best F1-score and MCC, while in dataset-2, the model trained with 16×16 patch size achieved the highest F1-score (69.42) and MCC (69.70). The patch size seems to be more relevant to dataset-2 since all the models trained with 16×16 patch size achieved the best results. In dataset-1, the SVM and KNN trained with the 16×16 patch size also had the best results; however, the best U-Net and RF model was trained with 128×128 and 32×32 patch size, respectively.

Comparing the results of both datasets, the models trained with dataset-1 achieved better results compared to the same algorithm over dataset-2. The U-Net 128,5 was the best overall model among both datasets. Similar to what was observed by Ghorbanzadeh et al. (2019a, b) with machine learning models trained in the same area, the topographical layers helped differentiate human settlement areas, which have identical spectral responses to landslides; however, the models generate more false-positive in the steeper areas. Visually evaluating the segmentation of each algorithms (as seen in Fig. 10), the U-Net segmentation is smoother and more continuous, with greater similarity in comparison to the manual annotations than with the other ML methods. SVM, KNN, and RF results show similar segmentation patterns and mistakes.

Discussion

The U-Net deep learning model achieved the best results in this study based on the metrics used to evaluate the models. However, the MCC and F1-score values were similar among all the models. The results highlight that U-Net can achieve robust results even with few training samples. However, since the machine and deep learning achieved similar accuracies, all the algorithms have similar behavior with a small dataset, and it is impossible to define a better algorithm based on the accuracy metrics. However, similarities between the manual annotations and the U-Net model results are noted in terms of landslide prediction smoothness and continuity, demonstrating better segmentation results than the other models. The models evaluated by Ghorbanzadeh et al. (2019a, b) in the same study area were trained with a bigger dataset composed of 3500 samples, which was augmented to 7000 samples. In that study, the CNN model achieved the best results with an F1-score that was

Fig. 4 (continued)

Landslide

Non Landslide

Fig. 5 **a** Landslide detection results using SVM model in sampled area in the test zone using dataset-1. **b** Landslide detection results using SVM model in sampled area in the test zone using dataset-2
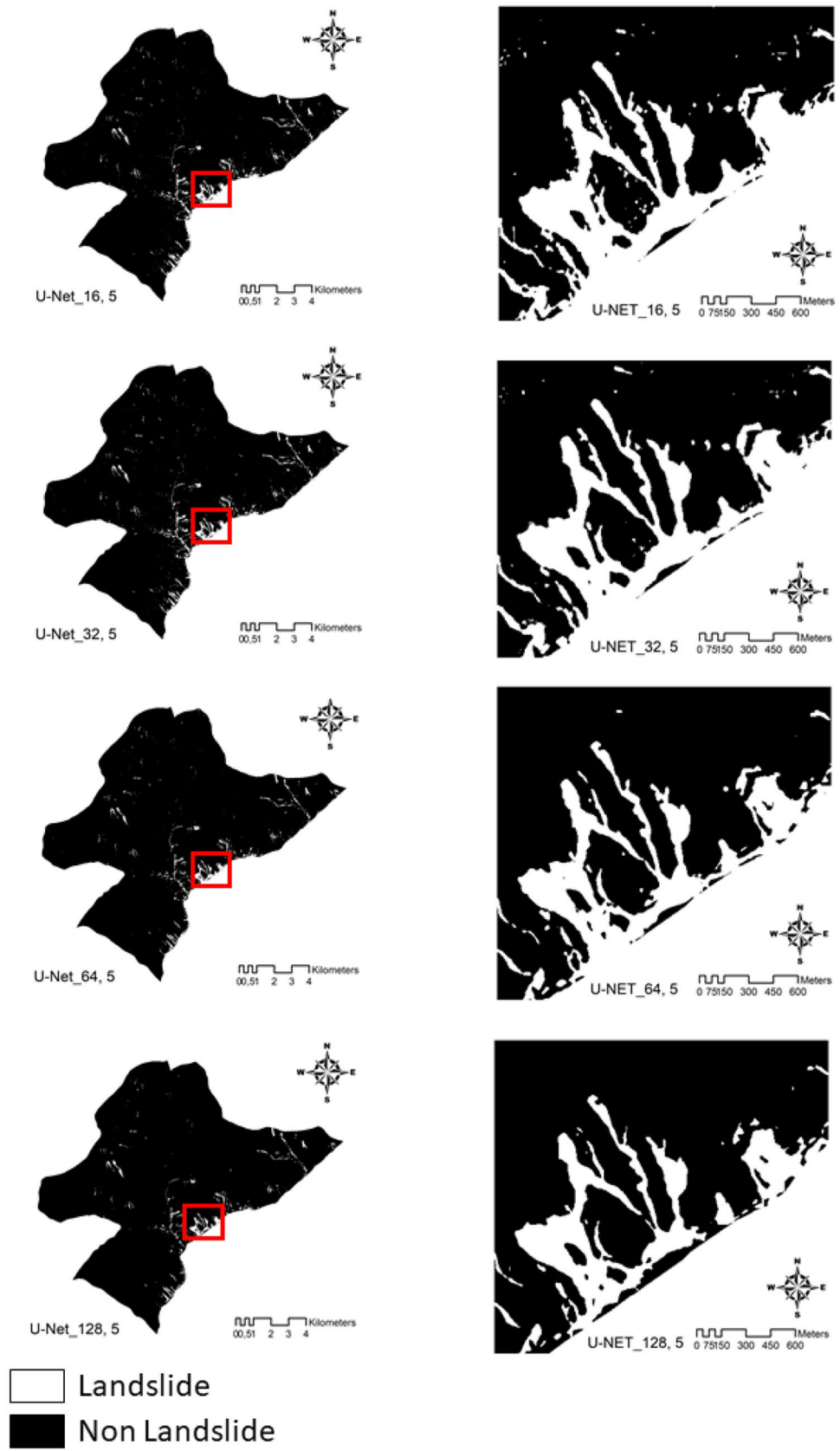
Fig. 5 (continued)

SVM_16, 7

SVM_16, 7

SVM_32, 7

SVM_32, 7

SVM_64, 7

SVM_64, 7

SVM_128, 7

SVM_128, 7
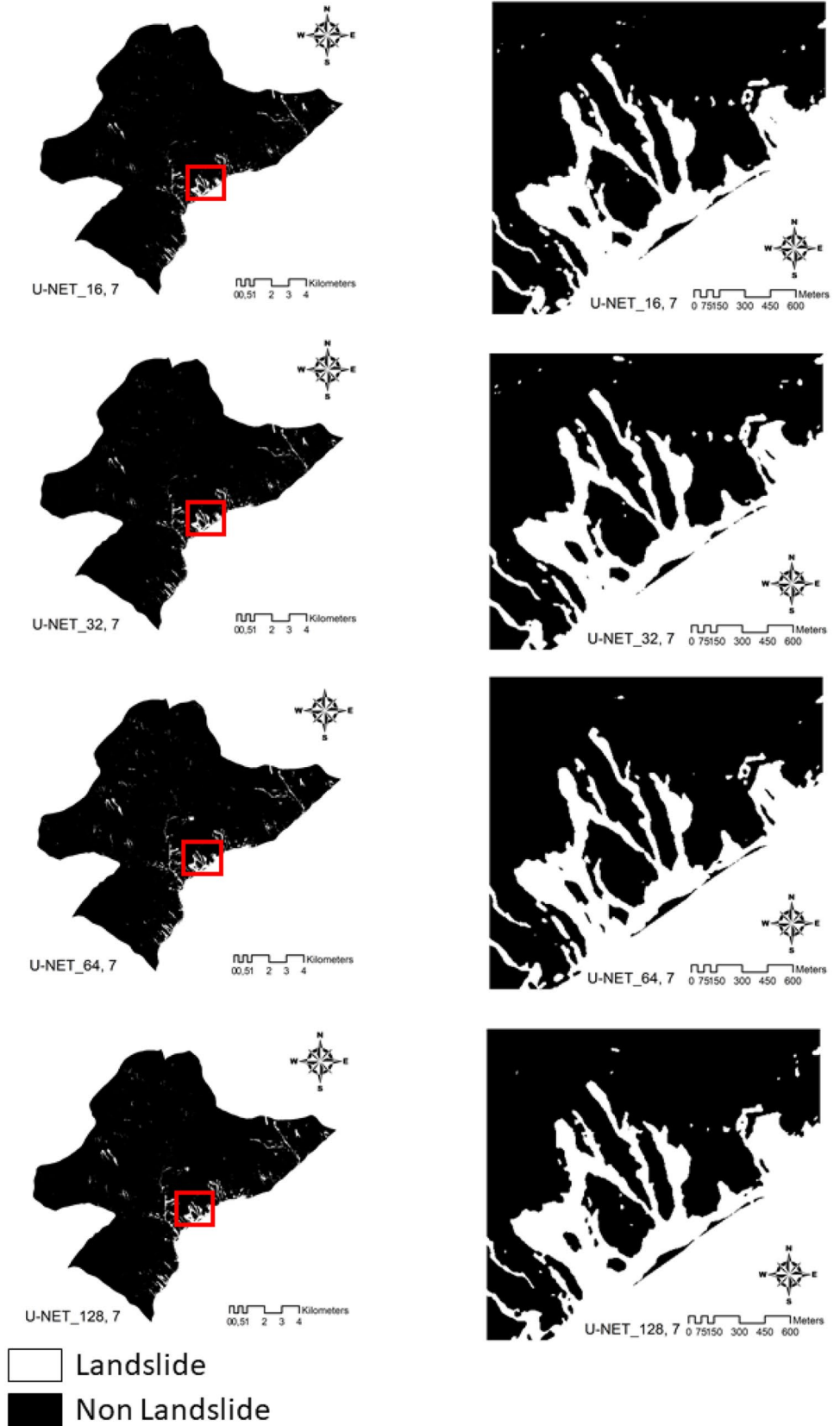
☐ Landslide
■ Non Landslide

Fig. 6 **a** Landslide detection results using KNN model in sampled area in the test zone using dataset-1. **b** Landslide detection results using KNN model in sampled area in the test zone using dataset-2.
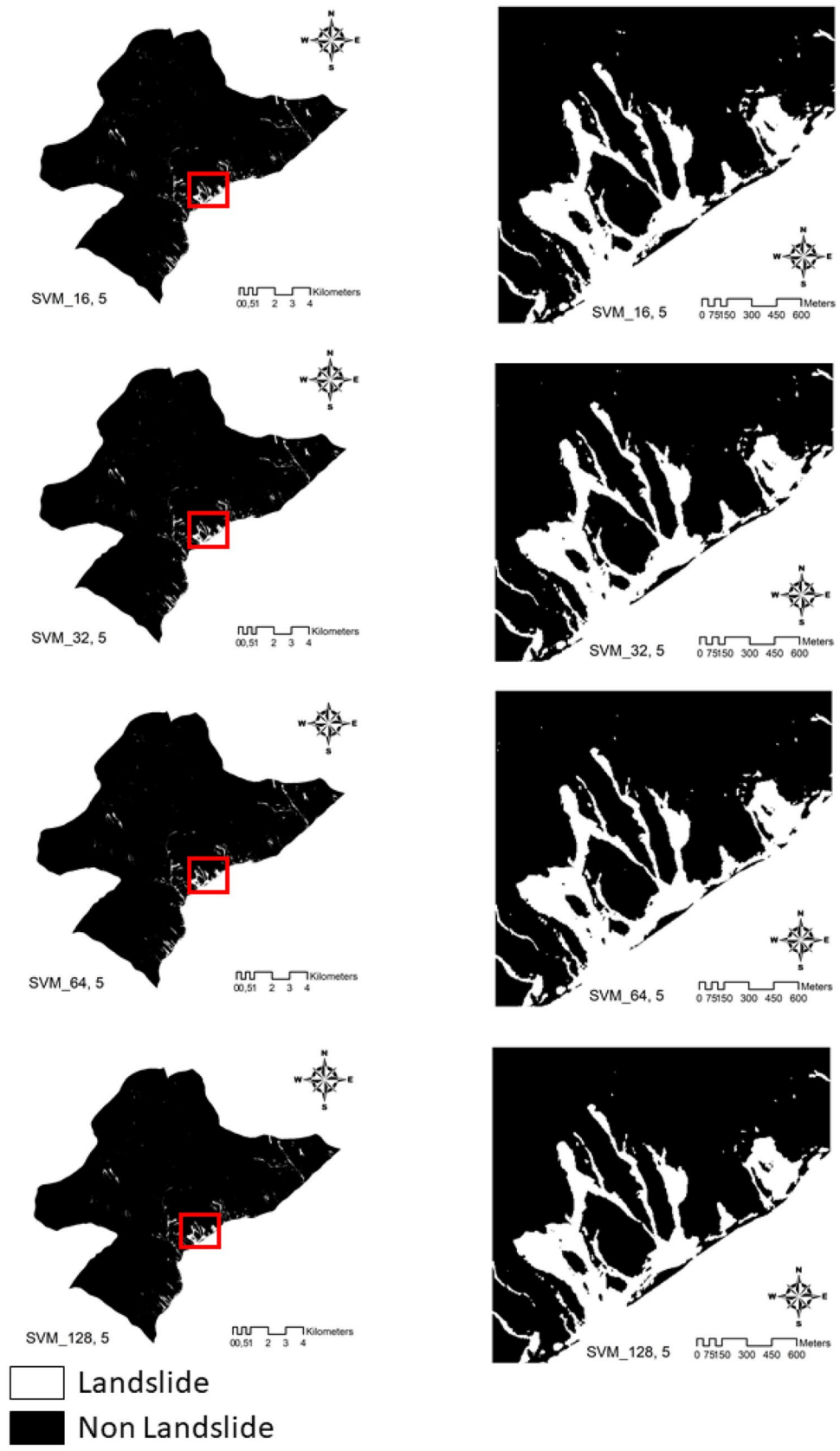

KNN_16, 5

KNN_32, 5

KNN_64 ,5

KNN_128, 5

☐ Landslide

■ Non Landslide

Fig. 6 (continued)

KNN_16, 7

KNN_16, 7

KNN_32, 7

KNN_32, 7

KNN_64, 7

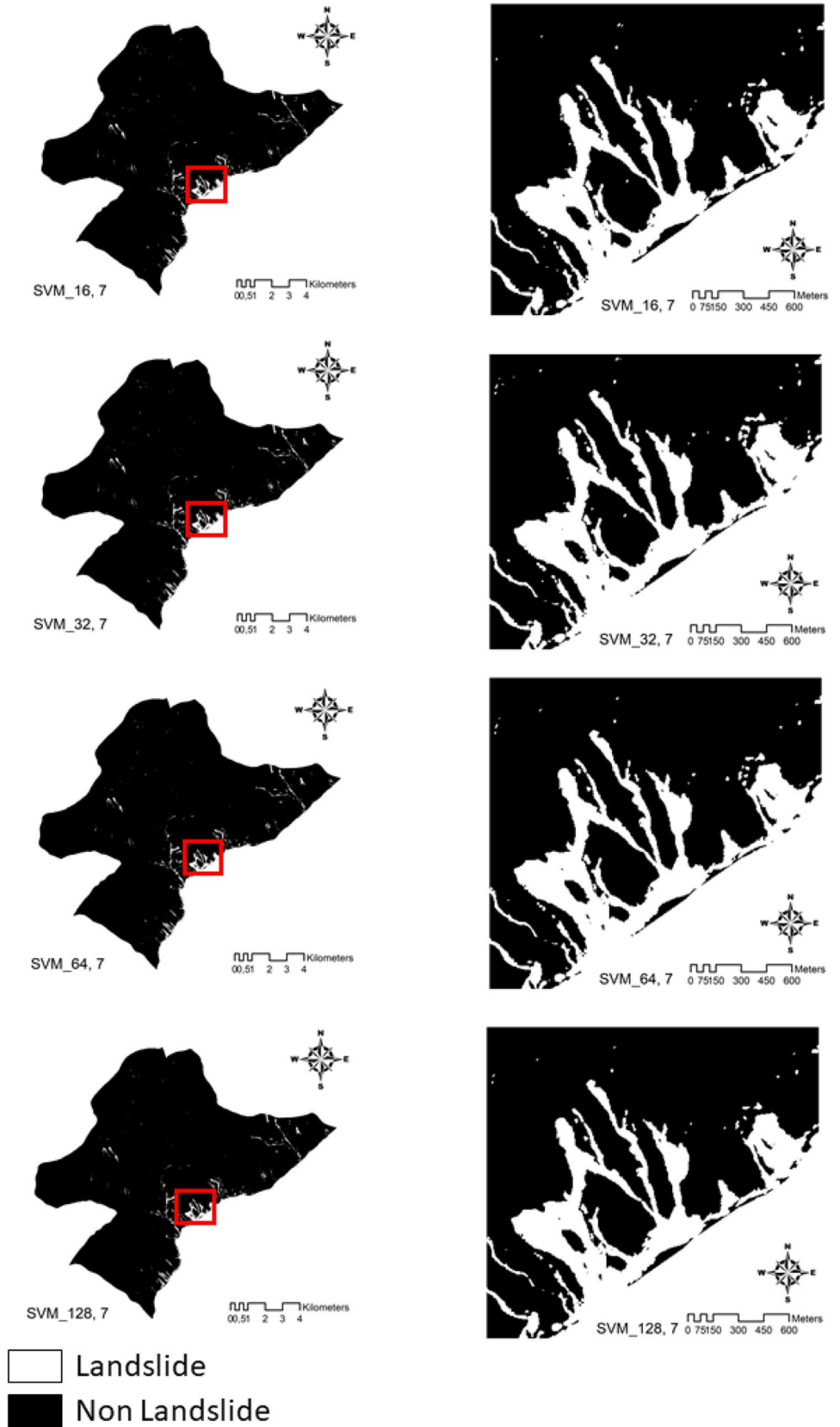KNN_64, 7
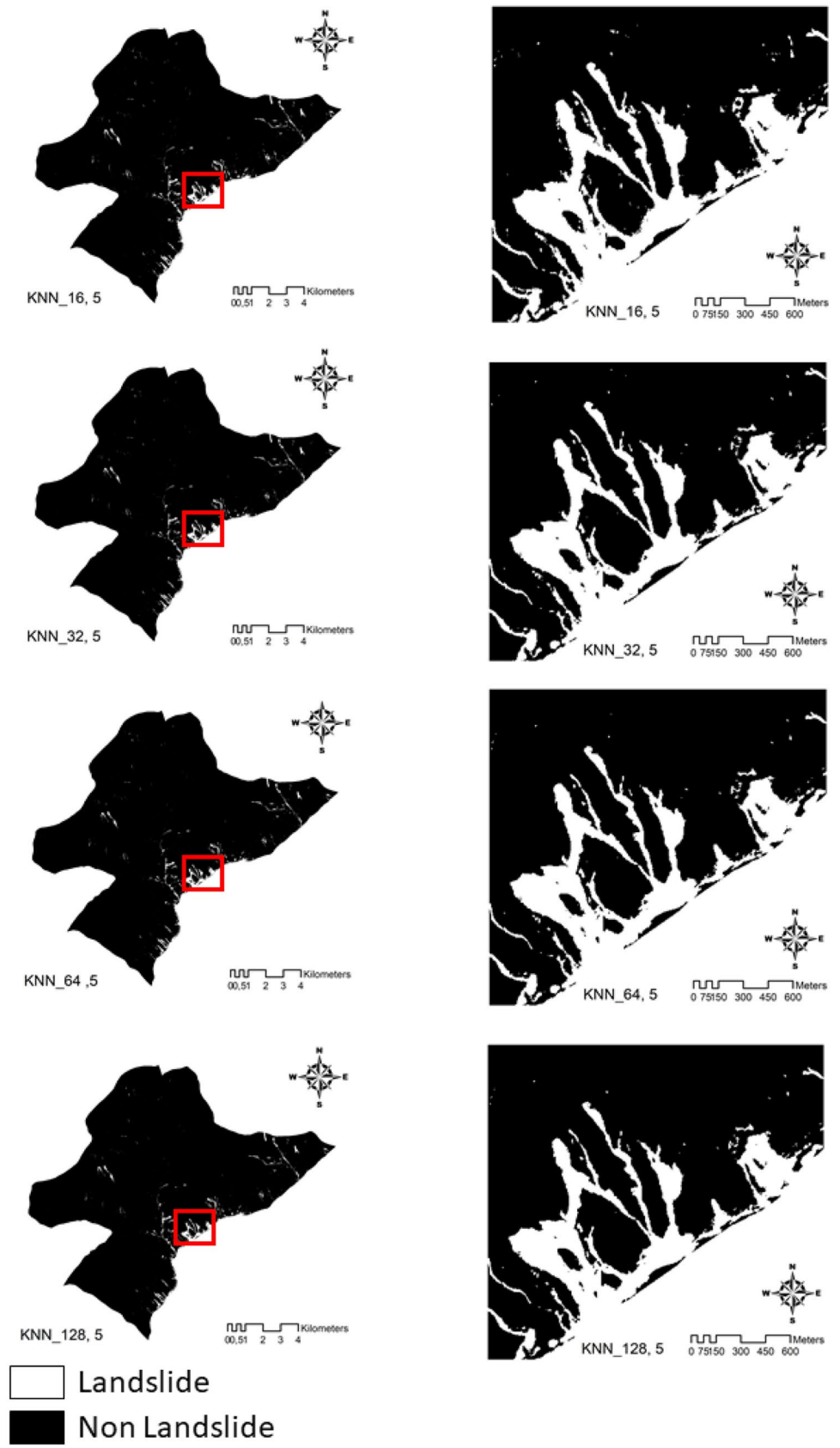
KNN_128, 7

KNN_128, 7

☐ Landslide
■ Non Landslide

Fig. 7 a Landslide detection results using RF model in sampled area in the test zone using dataset-1. **b** Landslide detection results using RF model in sampled area in the test zone using dataset-2.
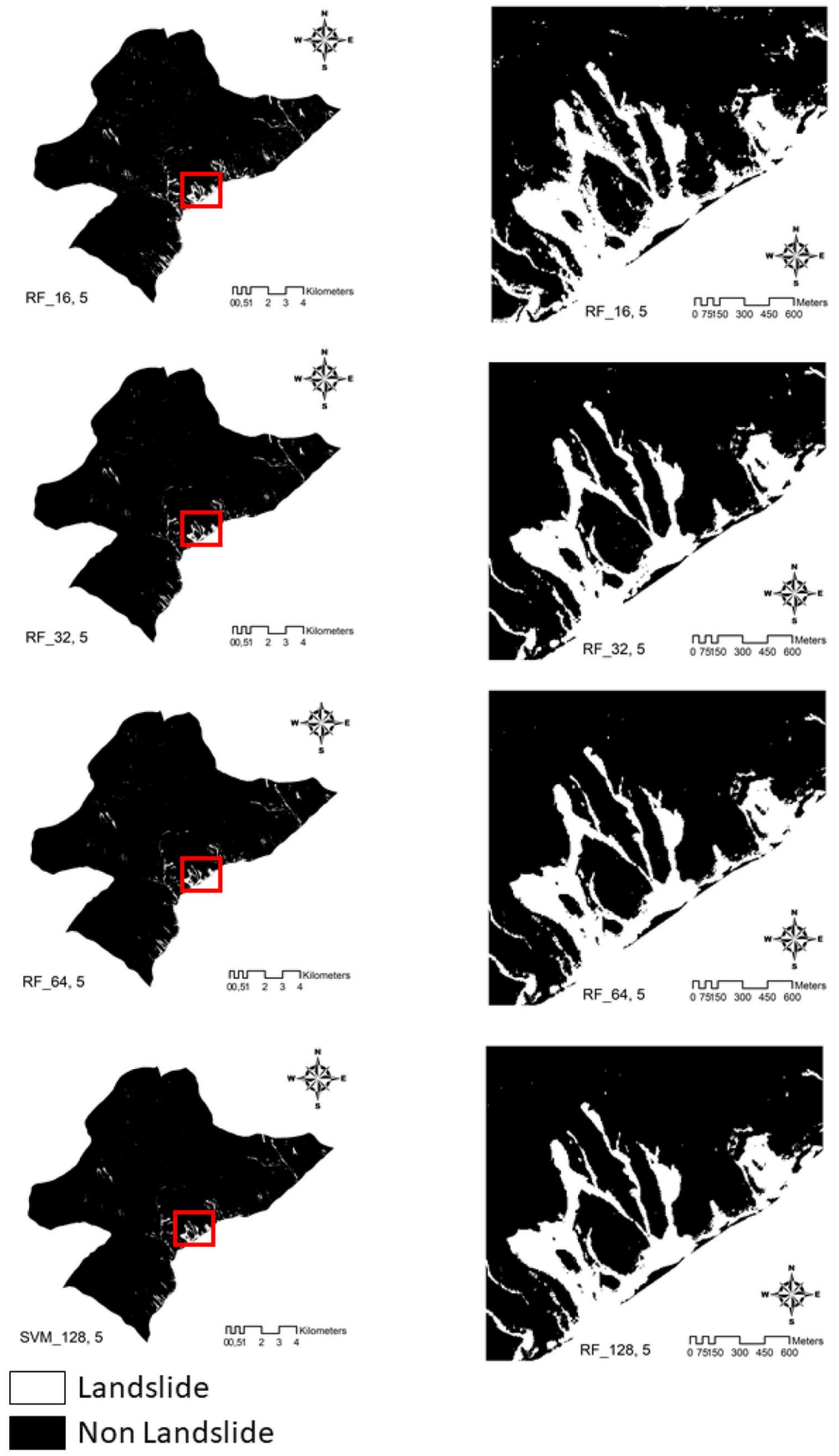
RF_16, 5   RF_16, 5

RF_32, 5   RF_32, 5

RF_64, 5   RF_64, 5

SVM_128, 5   RF_128, 5

☐ Landslide
■ Non Landslide

Fig. 7 (continued)

RF_16, 7

RF_32, 7

RF_64, 7

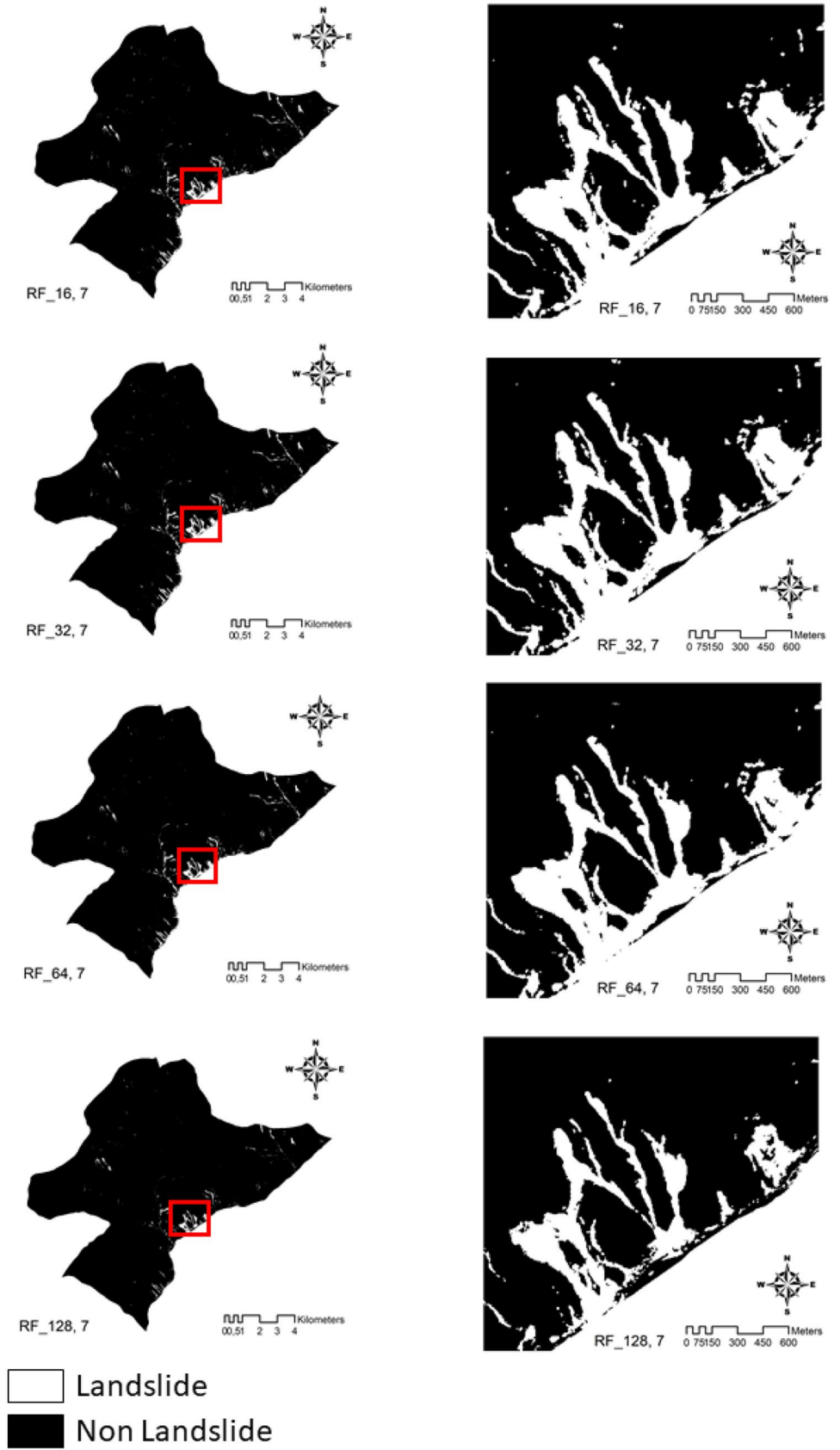RF_128, 7

☐ Landslide
■ Non Landslide

Fig. 8 Area of detected landslides using different machine learning and U-Net models against the manually interpreted ground truth (red color). **A** Dataset-1, **B** dataset-2
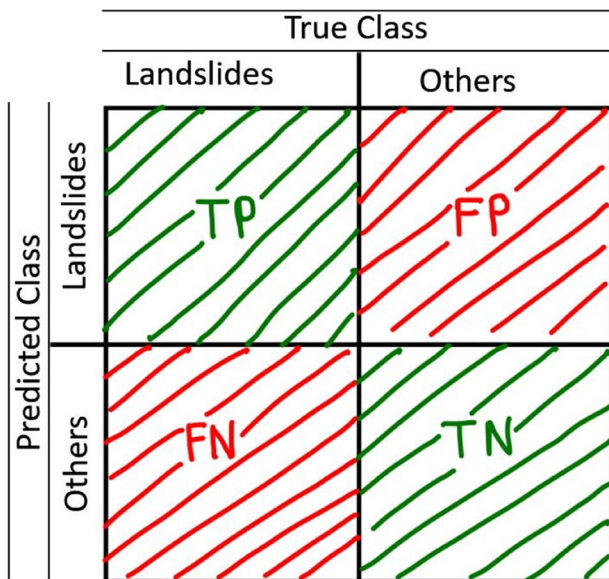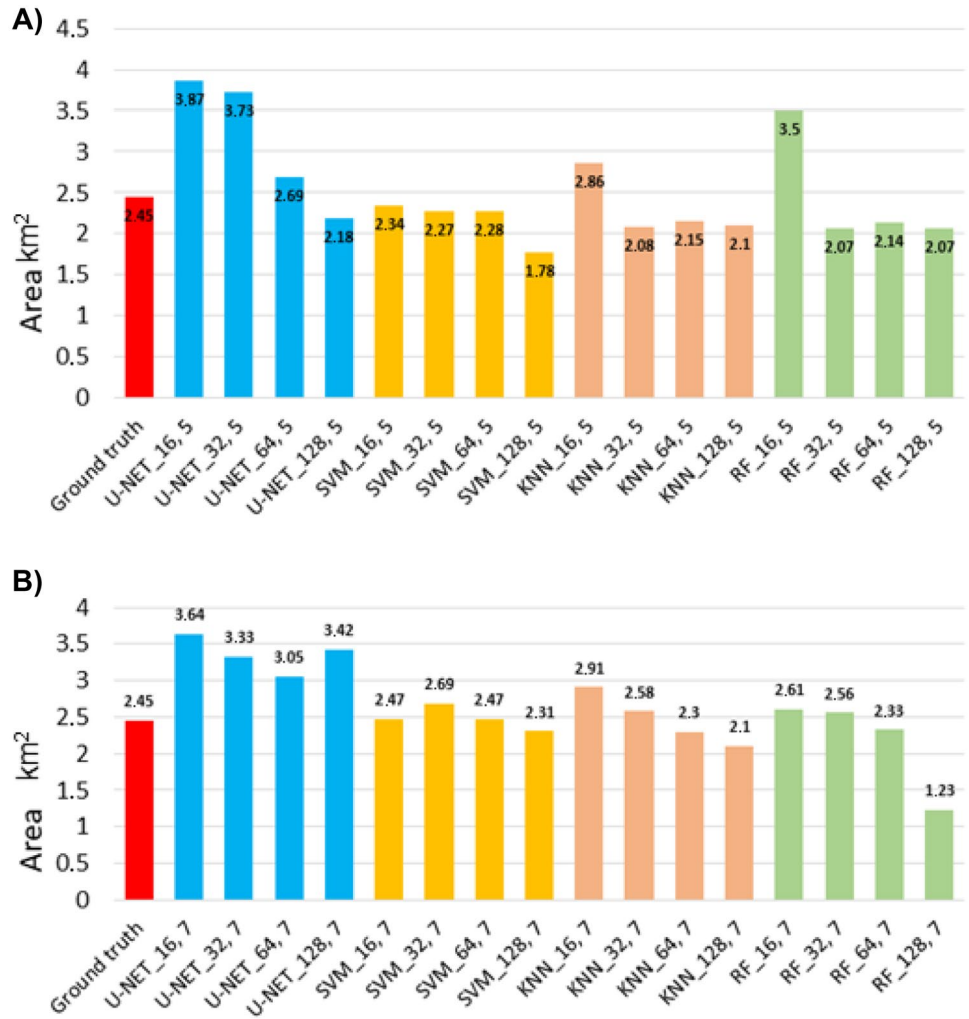

Fig. 9 Confusion matrix showing true class and predicted classes of landslides and other features and four different evaluation metrics

5.73% greater than the best machine learning model. The significant differences in the author's accuracy between the machine and deep learning models highlight the importance of the dataset size. In this study, despite the slightly higher accuracy achieved by the U-Net, the deep learning algorithms were computationally more expensive, needing a GPU (GeForce RTX 2060, 8 GB memory) for the training process, while the machine learning algorithms only used the CPU (Intel I7 10700 K).

The patch size is an important parameter to find the best algorithm since it affects the model's accuracy. The U-Net trained with the optical data showed a similar pattern to what was observed by Soares et al. (2020), where the U-Net models trained with smaller patches (32×32) yield a greater recall while the models trained with the bigger patches (128×128) achieved a greater precision. The models trained with bigger patches became more restrictive (made fewer false-positive errors) than the models trained with smaller patches. Nevertheless, this pattern was not observed in the U-Net models trained with the topographical dataset and on the results achieved by the machine learning models.

The topographical data does not improve the results of the models in this study. This may be related to the resampled DEM used and the samples. Since the dataset is composed of 174 samples, the

Table 2 The results of landslide detection in the study area based on the different ML and U-Net model for dataset-1; accuracies are stated as precision, recall, F1-measure, and MCC. The best values are in bold

Model	Precision	Recall	F1-Score	MCC
U-Net $_{16\_5}$	57.20	83.94	68.03	68.99
U-Net $_{32\_5}$	57.88	81.99	67.85	68.56
U-Net $_{64\_5}$	67.19	71.09	69.09	68.83
U-Net $_{128\_5}$	**76.59**	**66.38**	**71.12**	**71.06**
SVM $_{16\_5}$	75.75	65.95	70.51	70.43
SVM $_{32\_5}$	74.96	62.62	68.23	68.25
SVM $_{64\_5}$	72.43	60.65	66.02	66.00
SVM $_{128\_5}$	80.28	55.84	65.87	66.71
KNN $_{16\_5}$	67.79	71.77	69.73	69.47
KNN $_{32\_5}$	73.66	62.47	67.60	67.57
KNN $_{64\_5}$	73.36	61.39	66.84	66.84
KNN $_{128\_5}$	73.64	60.00	66.16	66.24
RF $_{16\_5}$	58.70	76.54	66.44	66.69
RF $_{32\_5}$	76.64	61.52	68.25	68.41
RF $_{64\_5}$	72.82	60.42	66.04	66.06
RF $_{128\_5}$	73.91	59.73	66.07	66.17

Table 3 The results of landslide detection in the study area based on the different ML and U-Net model for dataset-2; accuracies are stated as precision, recall, F1-measure, and MCC. The best values are in bold

Model	Precision	Recall	F1-Score	MCC
U-Net $_{16\_7}$	**61.46**	**79.74**	**69.42**	**69.70**
U-Net $_{32\_7}$	59.31	80.44	68.28	68.75
U-Net $_{64\_7}$	62.07	77.11	68.78	68.88
U-Net $_{128\_7}$	60.27	78.17	68.35	68.58
SVM $_{16\_7}$	72.20	66.52	69.24	69.04
SVM $_{32\_7}$	66.74	67.34	67.04	66.74
SVM $_{64\_7}$	68.80	63.19	65.88	65.64
SVM $_{128\_7}$	71.32	61.03	65.77	65.69
KNN $_{16\_7}$	66.24	72.58	69.27	69.05
KNN $_{32\_7}$	69.37	66.47	67.89	67.62
KNN $_{64\_7}$	70.25	63.29	66.59	66.39
KNN $_{128\_7}$	71.77	58.72	64.59	64.63
RF $_{16\_7}$	70.22	68.67	69.44	69.17
RF $_{32\_7}$	69.80	66.81	68.27	68.01
RF $_{64\_7}$	69.95	60.04	64.62	64.51

models were not exposed to various topographic features. Therefore, the pattern learned with the training samples may not represent the test area, and consequently, the results were worse. Moreover, the Hughes Phenomenon (Hughes 1968), also known as the Curse of Dimensionality in the field of machine learning, may also be related to the inferior results with the topographical dataset. Since the dataset with two extra topographical bands has a higher dimensionality, a greater number of samples are needed to improve the models' accuracy. The small number of samples used to train the models was not enough for the classifier to reliably classify the landslide areas; therefore, the classification performance degraded with the higher dimensional data. This phenomenon may also justify why the models trained with 16 × 16 patch size (smaller patch, with lower dimensionality) achieved the best results within this dataset.

The training and test area used in this study have landslides with similar spectral characteristics. Therefore, this may also explain the comparable results achieved with the machine and deep learning models. However, since machine learning algorithms are trained using a one-dimensional vector with pixel values, the spatial pattern of the landslides, such as the shapes, is not learned by those models. Consequently, it is expected that the deep learning method achieves better results in areas with different spectral characteristics than the machine learning algorithm because those models are trained with two-dimensional patches that keep the spatial information of the images. According to the literature, the U-Net like architectures achieve the best results for segmenting landslides in test areas with similar spectral characteristics to the training zones, and test areas with different spectral characteristics highlighting their generalization capacity and good accuracy on landslide segmentation (Qi et al. 2020; Prakasha et al. 2020; Soares et al. 2020; Yi et al. 2020; Prakasha et al. 2021).

Fig. 10 Enlarged maps of sub-area from the test zone. Landslide detection results are overlayed on the inventory data
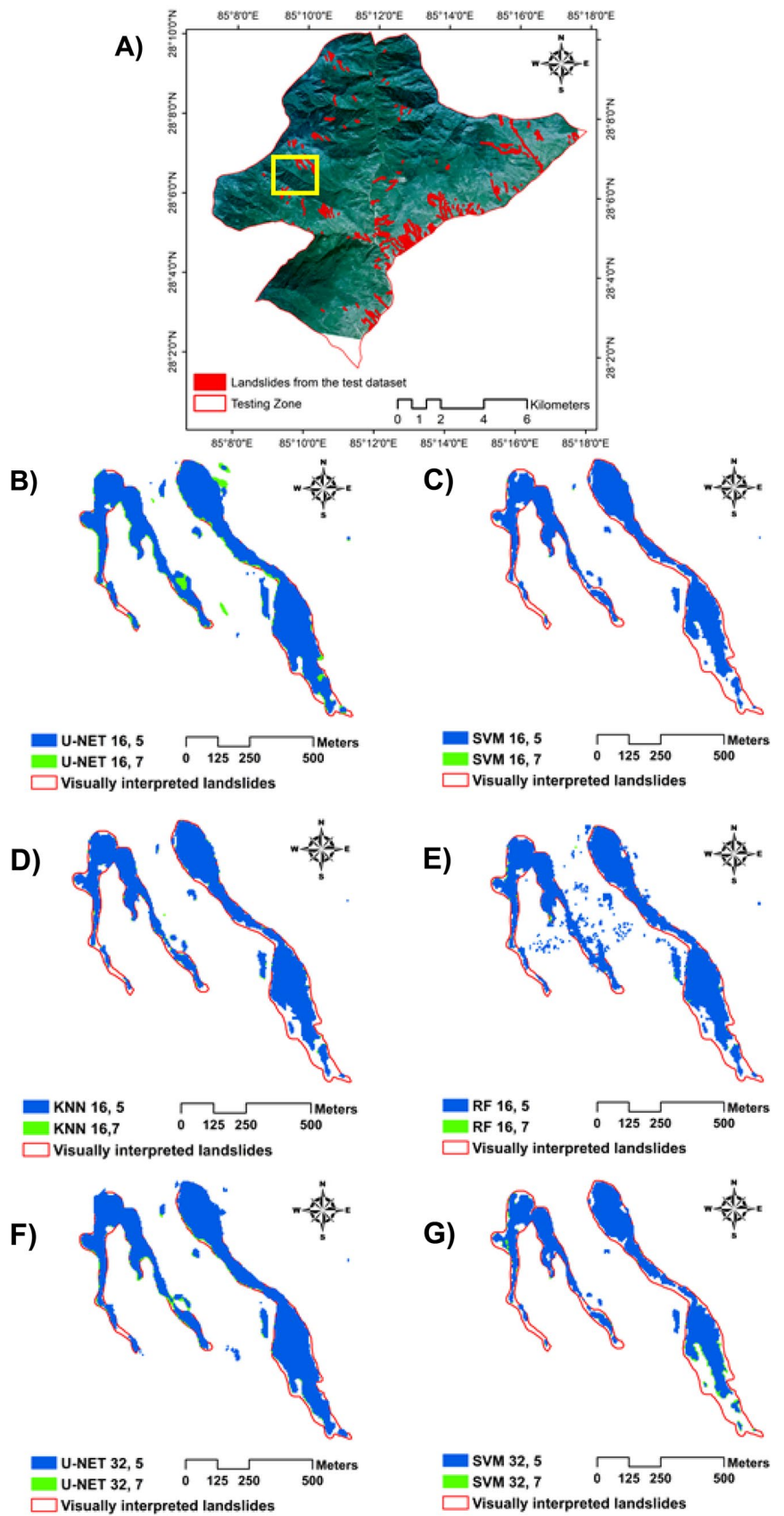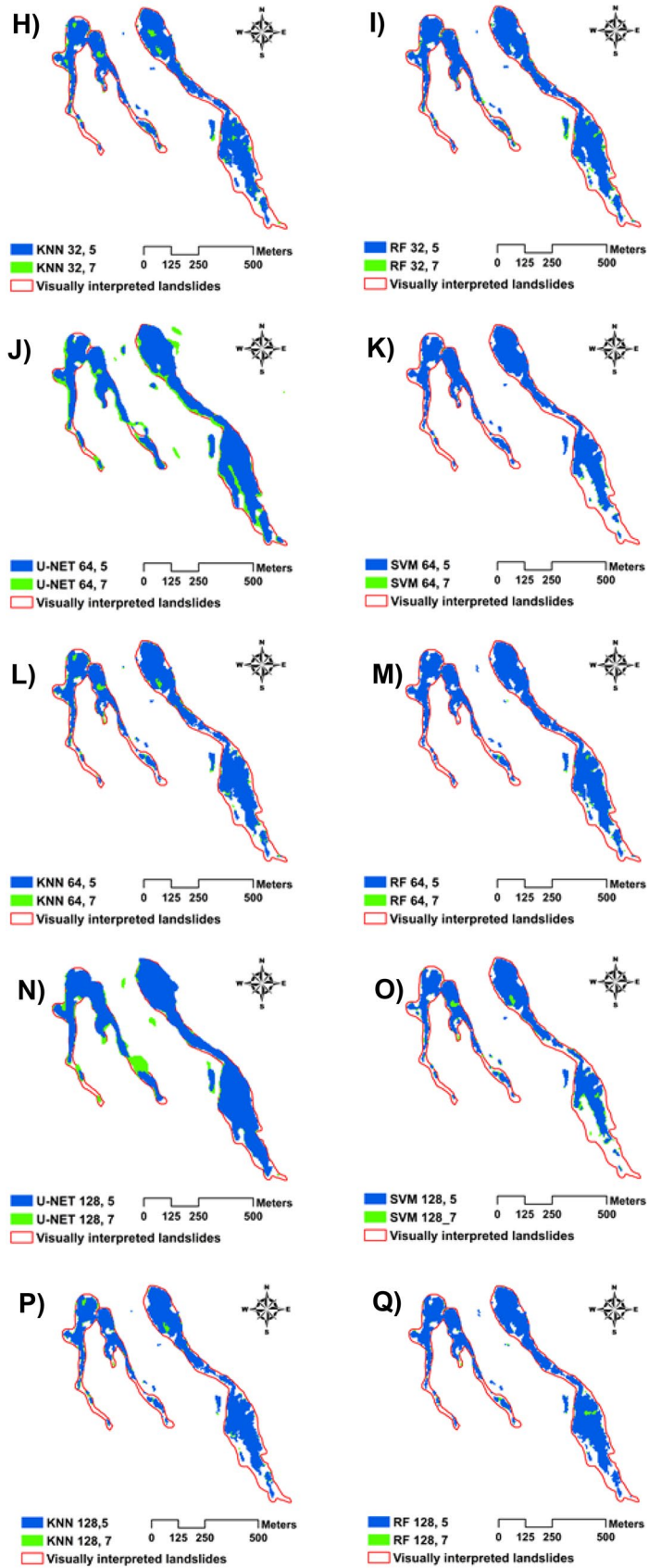
Fig. 10 (continued)

H)

KNN 32, 5
KNN 32, 7
Visually interpreted landslides
0 125 250 500 Meters

I)

RF 32, 5
RF 32, 7
Visually interpreted landslides
0 125 250 500 Meters

J)

U-NET 64, 5
U-NET 64, 7
Visually interpreted landslides
0 125 250 500 Meters

K)

SVM 64, 5
SVM 64, 7
Visually interpreted landslides
0 125 250 500 Meters

L)

KNN 64, 5
KNN 64, 7
Visually interpreted landslides
0 125 250 500 Meters

M)

RF 64, 5
RF 64, 7
Visually interpreted landslides
0 125 250 500 Meters

N)

U-NET 128, 5
U-NET 128, 7
Visually interpreted landslides
0 125 250 500 Meters

O)

SVM 128, 5
SVM 128_7
Visually interpreted landslides
0 125 250 500 Meters

P)

KNN 128,5
KNN 128, 7
Visually interpreted landslides
0 125 250 500 Meters

Q)

RF 128, 5
RF 128, 7
Visually interpreted landslides
0 125 250 500 Meters

Conclusions

This work evaluates different machine and deep learning model performances trained with small datasets and different patch sizes for landslide segmentation. The U-Net deep learning model achieved the best results on dataset-1 and dataset-2. However, all the models achieved similar MCC and F1-scores, highlighting that deep learning models achieve comparable results to machine learning algorithms with small datasets. The extra topographic features (slope and elevation) did not improve the models' results but yielded improved detection of false-positive such as built-up areas, an error in riverbeds. In this study, U-Net has slightly better results than other machine learning approaches. Although it can depend on the model architecture and the complexity of geographical features in the imagery, the U-Net model is still preliminary when considered for landslide detection. A reason for the U-Net model to perform better is because of the encoder-decoder and skip-connection structure of the model that preserves the structural integrity of the output results even with lower training data (Ronneberger et al. 2015). This exhibits the notion of actually using lesser training data, which is generally the case for new events, and can be then used in training and detecting landslides for newer events.

This study is one of the first efforts of using U-Net for landslide detection in the Himalayas. Nevertheless, U-Net has the potential to further improve automated landslide detection in the future as U-Net excels in producing good results as stated above in regard to the architecture structure but also that since the output is a segmentation result, we are provided with the information of the landslide boundary and the delineation of the landslide body as well. Further adjusting of the encoder part of the model, we can add deeper layers like Virtual Geometry Group (VGG) and Residual Neural Network (ResNet-50) (Simonyan and Zisserman 2014; He et al. 2016) to further improve the results and thereby detecting more landslides with fewer false positives as model complexity overall tends to overcome such artifacts.

The use of only spectral bands can be a limitation for landslide detection since geological and the degree of saturation of the soil directly affect the targets' spectral response. Therefore, areas with higher soil saturation may present darker colors while less saturated areas will have light colors. Moreover, rocks with different weathering conditions will show different spectral responses. Thus, to avoid algorithm misclassifications and improve the results, further studies need to use images covering a more comprehensive range of time and different seasons. This way, the models can learn and predict a broader range of spectral responses of the landslides and achieve better results.

Declarations

Conflict of interests   The authors declare no competing interests.

References

Bacha AS, Van der Werff H, Shafique M, Khan H (2020) Transferability of object-based image analysis approaches for landslide detection in the Himalaya Mountains of northern Pakistan. Int J Remote Sens 41:3390–3410. https://doi.org/10.1080/01431161.2019.1701725

Baldi P, Brunak S, Chauvin Y, Andersen CA, Nielsen H (2000) Assessing the accuracy of prediction algorithms for classification: an overview. Bioinformatics 16(5):412–424

Bui TA, Lee PJ, Lum KY, Chen CR, Shiu SH, Ieee (2019) Using BEMD in CNN to identify landslide in satellite image. In: Proceedings of 2019 International Conference on System Science and Engineering. Int Conf Sys Sci Eng 94–97

Chen GY, Li X, Gong WY, Xu H (2020) Recognition of the landslide disasters with extreme learning machine. Int J Comput Sci Eng 21:84–94. https://doi.org/10.1504/ijcse.2020.105215

Chen W, Li X, Wang Y, Chen G, Liu S (2014) Forested landslide detection using LiDAR data and the random forest algorithm: a case study of the Three Gorges. China Remote Sensing of Environment 152:291–301

Chen Z, Zhang Y, Ouyang C, Zhang F, Ma J (2018) Automated landslides detection for mountain cities using multi-temporal remote sensing imagery sensors (Basel) 18.https://doi.org/10.3390/s18030821

Cortes C, Vapnik V (1995) Support-Vector Networks Machine Learning 20:273–297

Dahal RK, Hasegawa S (2008) Representative rainfall thresholds for landslides in the Nepal Himalaya. Geomorphology 100:429–443

Dhital MR (2015) Geology of the Nepal Himalaya: regional perspective of the classic collided orogen. Springer

Ding A, Zhang Q, Zhou X, Dai B (2016) Automatic recognition of landslide based on CNN and texture change detection. In: 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC). IEEE 444–448

Fan X et al (2018) Coseismic landslides triggered by the 8th August 2017 Ms 7.0 Jiuzhaigou earthquake (Sichuan, China): factors controlling their spatial distribution and implications for the seismogenic blind fault identification Landslides 15:967–983.https://doi.org/10.1007/s10346-018-0960-x

Froude MJ, Petley DN (2016) Global fatal landslide occurrence from 2004 to 2018. Nat Hazard 18:2161–2181. https://doi.org/10.5194/nhess-18-2161-2018

Gariano SL, Guzzetti F (2016) Landslides in a Changing Climate Earth-Science Reviews 162:227–252. https://doi.org/10.1016/j.earscirev.2016.08.011

Ghorbanzadeh O, Blaschke T, Gholamnia K, Meena S, Tiede D, Aryal J (2019a) Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection remote sensing 11. https://doi.org/10.3390/rs1102019a6

Ghorbanzadeh O, Meena SR, Abadi HSS, Piralilou ST, Zhiyong L, Blaschke T (2021) Landslide mapping using two main deep-learning convolution neural network streams combined by the Dempster-Shafer model. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14:452–463. https://doi.org/10.1109/JSTARS.2020.3043836

Ghorbanzadeh O, Meena SR, Blaschke T, Aryal J (2019b) UAV-Based Slope Failure Detection Using Deep-Learning Convolutional Neural Networks Remote Sensing 11:2046

Ghorbanzadeh O, Tiede D, Wendt L, Sudmanns M, Lang S (2020) Transferable instance segmentation of dwellings in a refugee camp - integrating CNN and OBIA European. J Remote Sensing:1–14. https://doi.org/10.1080/22797254.2020.1759456

Guzzetti F, Mondini AC, Cardinali M, Fiorucci F, Santangelo M, Chang K-T (2012) Landslide Inventory Maps: New Tools for an Old Problem Earth-Science Reviews 112:42–66. https://doi.org/10.1016/j.earscirev.2012.02.001

Haigh M, Rawat JS (2011) Landslide causes: human impacts on a Himalayan landslide swarm Belgeo Revue belge de géographie 201–220

Hakan T, Luigi L (2020) Completeness index for earthquake-induced landslide inventories Engineering geology 264. https://doi.org/10.1016/j.enggeo.2019.105331

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition 770–778

Hölbling D, Füreder P, Antolini F, Cigna F, Casagli N, Lang S (2012) A Semi-Automated Object-Based Approach for Landslide Detection Validated by Persistent Scatterer Interferometry Measures and Landslide Inventories Remote Sensing 4:1310–1336. https://doi.org/10.3390/rs4051310

Hughes G (1968) On the mean accuracy of statistical pattern recognizers. IEEE Trans Inf Theory 14(1):55–63

Ji SP, Yu DW, Shen CY, Li WL, Xu Q (2020) Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks. Landslides 17:1337–1352. https://doi.org/10.1007/s10346-020-01353-2

Jin B, Ye P, Zhang X, Song W, Li S (2019) Object-oriented method combined with deep convolutional neural networks for land-use-type classification of remote sensing images. Journal of the Indian Society of Remote Sensing 47:951–965. https://doi.org/10.1007/s12524-019-00945-3

Khanal N, Watanabe T (2005) Landslide and debris flow hazards induced by heavy precipitation in Nepal. In: Proceedings of Himalaya International Symposium on Landslide Hazard in Orogenic Zone from the Himalaya to Island Arc in Asia 111–118

Lei T, Zhang Y, Lv Z, Li S, Liu S, Nandi AK (2019) Landslide inventory mapping from bitemporal images using deep convolutional neural networks. IEEE Geosci Remote Sens Lett 16:982–986. https://doi.org/10.1109/lgrs.2018.2889307

Liu P, Wei Y, Wang Q, Chen Y, Xie J (2020a) Research on post-earthquake landslide extraction algorithm based on improved U-Net model Remote sensing 12.https://doi.org/10.3390/rs12050894

Liu P, Wei YM, Wang QJ, Chen Y, Xie JJ (2020b) Research on Post-earthquake landslide extraction algorithm based on improved U-Net model Remote sensing 12.https://doi.org/10.3390/rs12050894

Liu S, Qi Z, Li X, Yeh AG-O (2019) Integration of convolutional neural networks and object-based post-classification refinement for land use and land cover mapping with optical and SAR data Remote sensing 11:690

Lu H, Ma L, Fu X, Liu C, Wang Z, Tang M, Li NW (2020) landslides information extraction using object-oriented image analysis paradigm based on deep learning and transfer learning Remote sensing 12. https://doi.org/10.3390/rs12050752

Marjanovic M, Bajat B, Kovacevic M (2009) Landslide susceptibility assessment with machine learning algorithms. In: 2009 International Conference on Intelligent Networking and Collaborative Systems. IEEE 273–278

Martha TR, Kerle N, Jetten V, van Westen CJ, Kumar KV (2010) Characterising spectral, spatial and morphometric properties of landslides for semi-automatic detection using object-oriented methods. Geomorphology 116:24–36.https://doi.org/10.1016/j.geomorph.2009.10.004

Meena S, Chauhan A, Bhuyan K, Singh RP (2021a) Impact of the Chamoli disaster on flood plain and water quality along the Himalayan rivers. Paper presented at the EGU General Assembly

Meena SR, Ghorbanzadeh O, van Westen CJ, Nachappa TG, Blaschke T, Singh RP, Sarkar R (2021b) Rapid mapping of landslides in the Western Ghats (India) triggered by 2018 extreme monsoon rainfall using a deep learning approach Landslides.https://doi.org/10.1007/s10346-020-01602-4

Meena SR, Tavakkoli Piralilou S (2019) Comparison of earthquake-triggered landslide inventories: a case study of the 2015 Gorkha earthquake. Nepal Geosciences 9:437

Moosavi L, Mahyuddin N, Ab Ghafar N, Ismail MA (2014) Thermal Performance of Atria: an Overview of Natural Ventilation Effective Designs Renewable and Sustainable Energy Reviews 34:654–670

Mountrakis G, Im J, Ogole C (2011) Support vector machines in remote sensing: a review. ISPRS J Photogramm Remote Sens 66:247–259

Pawłuszek K, Borkowski A (2016) Landslides identification using airborne laser scanning data derived topographic terrain attributes and support vector machine classification. In: The international archives of the photogrammetry, remote sensing and spatial information sciences, XXIII ISPRS Congress

Petley D (2012) Global patterns of loss of life from landslides. Geology 40:927–930

Prakash N, Manconi A, Loew S (2020) Mapping landslides on EO data: performance of deep learning models vs. traditional machine learning models Remote sensing 12.https://doi.org/10.3390/rs12030346

Prakash N, Manconi A, Loew S (2021) A new strategy to map landslides with a generalized convolutional neural network. Sci Rep 11(1):1–15

Qi W, Wei M, Yang W, Xu C, Ma C (2020) Automatic mapping of landslides by the resU-Net Remote sensing 12. https://doi.org/10.3390/rs12152487

Reichenbach P, Rossi M, Malamud BD, Mihir M, Guzzetti F (2018) A Review of Statistically-Based Landslide Susceptibility Models Earth-Science Reviews 180:60–91. https://doi.org/10.1016/j.earscirev.2018.03.001

Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. International Conference on Medical image computing and computer-assisted intervention. Springer, pp 234–241

RapidEye AG (2011) RapidEye satellite imagery product specifications. Version 3.2. RapidEye AG, product manual

Sameen MI, Pradhan B (2019) Landslide detection using residual networks and the fusion of spectral and topographic information IEEE. Access 7:114363–114373. https://doi.org/10.1109/access.2019.2935761

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556

Shi W, Zhang M, Ke H, Fang X, Zhan Z, Chen S (2020) Landslide recognition by deep convolutional neural network and change detection IEEE transactions on geoscience and remote sensing 1–19. https://doi.org/10.1109/TGRS.2020.3015826

Soares LP, Dias HC, Grohmann CH (2020) Landslide segmentation with U-Net: Evaluating different sampling methods and patch sizes. arXiv preprint arXiv:2007.06672

Su Z, Chow JK, Tan PS, Wu J, Ho YK, Wang Y-H (2020) Deep convolutional neural network–based pixel-wise landslide inventory mapping Landslides.https://doi.org/10.1007/s10346-020-01557-6

Thapa P, Dhital M (2000) Landslide and debris flows of 19–21 July 1993 in the Agra Khola watershed of central. Nepal Jour Nepal Geol Soc 21:5–20

Upreti BN, Dhital M (1996) Landslide studies and management in Nepal

Wang Y, Wu X, Chen Z, Ren F, Feng L, Du Q (2019) Optimizing the predictive ability of machine learning methods for landslide susceptibility mapping using SMOTE for Lishui City in Zhejiang Province, China. Int J Environ Res Public Health 16. https://doi.org/10.3390/ijerph16030368

Whitworth M, Giles D, Murphy W Identification of landslides in clay terrains using Airborne Thematic Mapper (ATM) multispectral imagery.

In: Remote Sensing for Environmental Monitoring, GIS Applications, and Geology, 2002. Int Soc Optics and Photonics 216–224

Ye CM et al (2019) Landslide detection of hyperspectral remote sensing data based on deep learning with constrains. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12:5047–5060. https://doi.org/10.1109/jstars.2019.2951725

Yi Y, Zhang Z, Zhang W, Jia H, Zhang J (2020) Landslide susceptibility mapping using multiscale sampling strategy and convolutional neural network: A case study in Jiuzhaigou region. Catena 195:104851

Yu B, Chen F, Xu C (2020) Landslide detection based on contour-based deep learning framework in case of national scale of Nepal in 2015 Computers & geosciences 135.https://doi.org/10.1016/j.cageo.2019.104388

Yu H, Ma Y, Wang LF, Zhai YS, Wang XQ, Ieee (2017) A landslide intelligent detection method based on CNN and RSG_R. 2017 Ieee Int Conf Mechatronics and Automation

Zhang P, Ke Y, Zhang Z, Wang M, Li P, Zhang S (2018) Urban Land Use and Land Cover Classification Using Novel Deep Learning Models Based on High Spatial Resolution Satellite Imagery Sensors 18:3717

Sansar Raj Meena (✉) · **Cees** van **Westen**
Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente, Enschede, The Netherlands
Email: s.r.meena@utwente.nl

Sansar Raj Meena · Kushanav Bhuyan · Mario Floris · Filippo Catani
Machine Intelligence and Slope Stability Laboratory, Department of Geosciences, University of Padova, Padova, Italy
Email: s.r.meena@utwente.nl

Lucas Pedrosa Soares
Institute of Geosciences, University of São Paulo (IGc-USP), São Paulo 05508-080, Brazil

Carlos H. Grohmann
Institute of Energy and Environment, University of São Paulo (IEE-USP), São Paulo 05508-010, Brazil

Lucas Pedrosa Soares · Carlos H. Grohmann
Spatial Analysis and Modelling Lab (SPAMLab) -, São Paulo, Brazil

Ramesh P. Singh
School of Life and Environmental Sciences, Schmid College of Science and Technology, Chapman University One University Drive, Orange, CA, USA

7.4 Feature-based constraint deep CNN method for mapping rainfall- induced landslides in remote regions with mountainous terrain: An application to Brazil

Feature-Based Constraint Deep CNN Method for Mapping Rainfall-Induced Landslides in Remote Regions With Mountainous Terrain: An Application to Brazil

Guosen Xu , Yi Wang , *Member, IEEE*, Lizhe Wang , *Fellow, IEEE*, Lucas Pedrosa Soares ,
and Carlos H. Grohmann

Abstract—Landslides have caused tremendous damage to human lives and property safety. However, the complex environment of mountain landslides and the vegetation coverage around landslides make it difficult to identify landslides quickly and efficiently using high-resolution images. To address this challenge, this article presents a feature-based constraint deep U-Net (FCDU-Net) method to detect rainfall-induced mountainous landslides. Usually, the vegetation in the landslide area is severely damaged, and the vegetation coverage can indirectly reflect the spatial extent of the landslide. Meanwhile, the texture features of high-resolution images can characterize the surface environment of landslide hazards to a certain extent. We first introduce auxiliary features of normalized difference vegetation index and gray-level co-occurrence matrix into the proposed method to further improve the detection performance. Then, to minimize the information redundancy of these features and the image, we combine Relief-F and Deep U-Net to screen the optimal features to effectively identify accurate and detailed landslide boundaries. Compared with traditional semantic segmentation methods, the FCDU-Net method can capture fine-grained details in high-resolution images and produce more accurate segmentation results. We conducted experiments by applying the proposed method and other most popular semantic segmentation methods to a high-resolution RapidEye image in Rio de Janeiro, Brazil. The results demonstrate that the FCDU-Net method can achieve better landslide detection results than the other semantic segmentation methods, and the evaluation measures of Precision, F1 score, and mean Intersection-over-Union are as high as 88.87%, 81.17%, and 83.19%, respectively. Furthermore, we quantitatively analyze the effect of the convolution input window size on the performance of FCDU-Net in detecting landslides. We believe that FCDU-Net can serve as a reliable tool for fast and accurate regional landslide hazard surveys.

Index Terms—Auxiliary features, deep convolutional neural networks (DCNNs), feature selection, high-resolution image, landslide detection.

I. Introduction

AS A common geological disaster, landslides cause immeasurable damage to infrastructure and seriously threaten the safety of human life and property [1]–[3]. Therefore, efficiently and accurately locating of landslide areas is crucial for emergency rescue and disaster warning. Landslide inventory data is the main resource for landslide research at different scales [4]. The training and testing of landslide prediction begins with the creation of a landslide inventory map [5]. Traditional landslide inventory mapping methods mainly rely on field survey and visual interpretation of aerial images [6], but it requires personnel with professional knowledge to complete it, wastes a lot of manpower and material resources, and cannot meet the needs of efficient and accurate positioning of landslides.

Remote sensing data can cover hundreds of square kilometers, providing rich image data for disaster observation. Optical and radar remote sensing images have been widely used to observe the landscape changes of landslides [7]–[12]. Traditional image processing techniques mainly detect landslides on remote sensing images through statistical methods [10] and semi-automatic methods [13]. However, these methods need to obtain the image features of the landslide area in advance by artificial means, and also require a lot of parameter modulations, and the process is complicated.

Satellite images are significant for identifying landslides following large-scale events, especially when the disaster occurs in remote or hard-to-reach areas [14]. Generally, landslides can be automatically identified in aerial photographs, high-resolution

Guosen Xu is with the Institute of Geophysics and Geomatics, China University of Geosciences, Wuhan 430074, China (e-mail: guosenn@163.com).

Yi Wang is with the Institute of Geophysics and Geomatics, and State Key Laboratory of Biogeology and Environmental Geology, China University of Geosciences, Wuhan 430074, China (e-mail: cug.yi.wang@gmail.com).

Lizhe Wang is with the Institute of Computer Science and State Key Laboratory of Biogeology and Environmental Geology, China University of Geosciences, Wuhan 430074, China (e-mail: lizhe.wang@gmail.com).

Lucas Pedrosa Soares is with the Institute of Geosciences, University of São Paulo, São Paulo 05508-080, Brazil (e-mail: lpsoares@usp.br).

Carlos H. Grohmann is with the Institute of Energy and Environment, University of São Paulo, São Paulo 05508-010, Brazil (e-mail: guano@usp.br).

Digital Object Identifier 10.1109/JSTARS.2022.3161383

and very-high-resolution remote sensing images by using pixelwise identification methods and object-based information extraction (OBIE) methods [15]. Among them, pixelwise methods can overcome the shortcomings of human interpretation, and can be mainly separated into two main categories: supervised learning and unsupervised learning. The former includes discriminant analysis, feature analysis, maximum likelihood analysis, support vector machine, etc., and the latter includes iterative self-organizing data analysis technique algorithm and K-means. These pixelwise methods were used to identify landslides in Taiwan and compared with visual interpretation results [16]. After discussing six different image fusion methods, Marcelino *et al.* [17] proved that the intensity-hue-saturation technique can better preserve the information of the original image and used it to identify landslide scars. In addition, the change detection technique allows the extraction of landslides from images at different time phases [18]–[25]. The pixelwise methods only exploit the spectral features of different materials in the image without considering the geometric and contextual information in the image. Therefore, these methods are prone to "pepper and salt" noise in the resultant map. However, OBIE-based methods can consider multiple features, such as spectral (pixel value, hue, color, etc.), spatial (landslide area size, shape, contour, etc.), and background features [26], which are widely used for landslide detection using airborne and satellite images [27], [28]. For example, Martin *et al.* [29] took the image segmentation result as the basic unit and established landslide recognition constraints based on shape, texture, and adjacent features, and the classification accuracy reached 65%. Based on FORMOSAT-2 images and DEM data, Rau *et al.* [30] identified landslides in the mountainous areas of central and southern Taiwan through multilevel segmentation and hierarchical semantic network, and achieved good experimental results. Meanwhile, conventional classification methods such as support vector machine [31] and random forest [28] have also been introduced into the OBIE-based landslide detection, and their effectiveness has been verified. However, some empirical parameters of OBIE rely too much on experts and the degree of automation is low. Thus, there are still certain limitations in practical applications [32], [33].

Convolutional neural networks (CNNs) can effectively capture abstract features from the original image [34]. In recent years, CNNs have attracted extensive attention and applications in the fields of large-scale image recognition, target detection, and semantic segmentation due to their advantages of local receptive, weight sharing, and connection sparsity [35]–[38]. However, only a few articles have used CNNs for landslide detection based on remote sensing image [2]. Ding *et al.* [39] first used texture and spectral features to remove the interference of background features such as buildings, water bodies, and vegetation, and then used CNNs to capture remote sensing image information before and after the landslide and calculated the Euclidean distance of the two features to determine whether there is a landslide on the image. Yu *et al.* [40] first screened out candidate images with landslides from remote sensing images through a shallow CNN, and then used a region growing algorithm to determine landslide boundaries and centers. Ghorbanzadeh *et al.* [1] performed different numbers of convolutions

and different levels of feature fusion on remote sensing images of southern Nepal. To further reduce the influence of background on the image, Ji *et al.* [2] developed a deep convolutional neural network (DCNN) for end-to-end landslide target detection. Yi *et al.* [41] constructed a cascaded end-to-end DCNN to intelligently detect earthquake-triggered landslides. Qin *et al.* [42] applied distant domain transfer learning and traditional CNNs for landslide detection.

In summary, traditional feature extraction methods and CNNs have been used for landslide detection in high-resolution images. However, only a few articles have used deep learning techniques to detect landslides in areas with complex terrain, especially those triggered by extreme weather such as heavy rain. We believe that the task of detecting landslides in mountainous areas caused by heavy rainfall still faces the following challenges. First, the background of remote sensing images with mountainous terrain is very complex, including quarries, terraces, slopes, and riverbeds, which may be similar in characteristics to landslides and easily interfere with detecting them. Second, landslides usually appear at different scales on remote sensing images, and the length of landslides may vary from a few meters to several kilometers. Finally, there are obvious seasonal changes in remote sensing images of mountainous areas, and the specific characteristics of coverage in different seasons may make it difficult to identify landslides.

To fill a gap in the previous articles, we present a feature-based constraint deep U-Net (FCDU-Net) method to map rainfall-induced landslides in remote areas with mountainous terrain. The main contributions are summarized as follows:

1) We only need to acquire postdisaster optical remote sensing imagery and process landslide data through data augmentation strategies, without inputting other data such as topographic factors and multitemporal images.

2) Deep U-Net (DU-Net) is used as a prediction model for landslide detection. Since it combines the advantages of U-Net and DenseNet, it can capture fine-grained details in high-resolution images and produce more accurate semantic segmentation results than most popular networks.

3) The FCDU-Net method combines auxiliary features with the original image to improve the identification accuracy of landslides. Usually, the vegetation in the landslide area is severely damaged, and the vegetation coverage can indirectly reflect the spatial extent of the landslide. Meanwhile, the texture features of high-resolution images can characterize the surface environment of landslide hazards to a certain extent. On this basis, normalized difference vegetation index (NDVI) and gray-level co-occurrence matrix (GLCM) are input into the network as auxiliary features. Furthermore, to minimize the information redundancy of these features and the image, we combine Relief-F and DU-Net to screen the optimal features for subsequent processes, and effectively identify accurate and detailed landslide boundaries.

The rest of this article is organized as follows. Section II introduces the study area. Section III describes the auxiliary features used in this article and the modified DU-Net method. Section IV analyzes landslide detection results of different
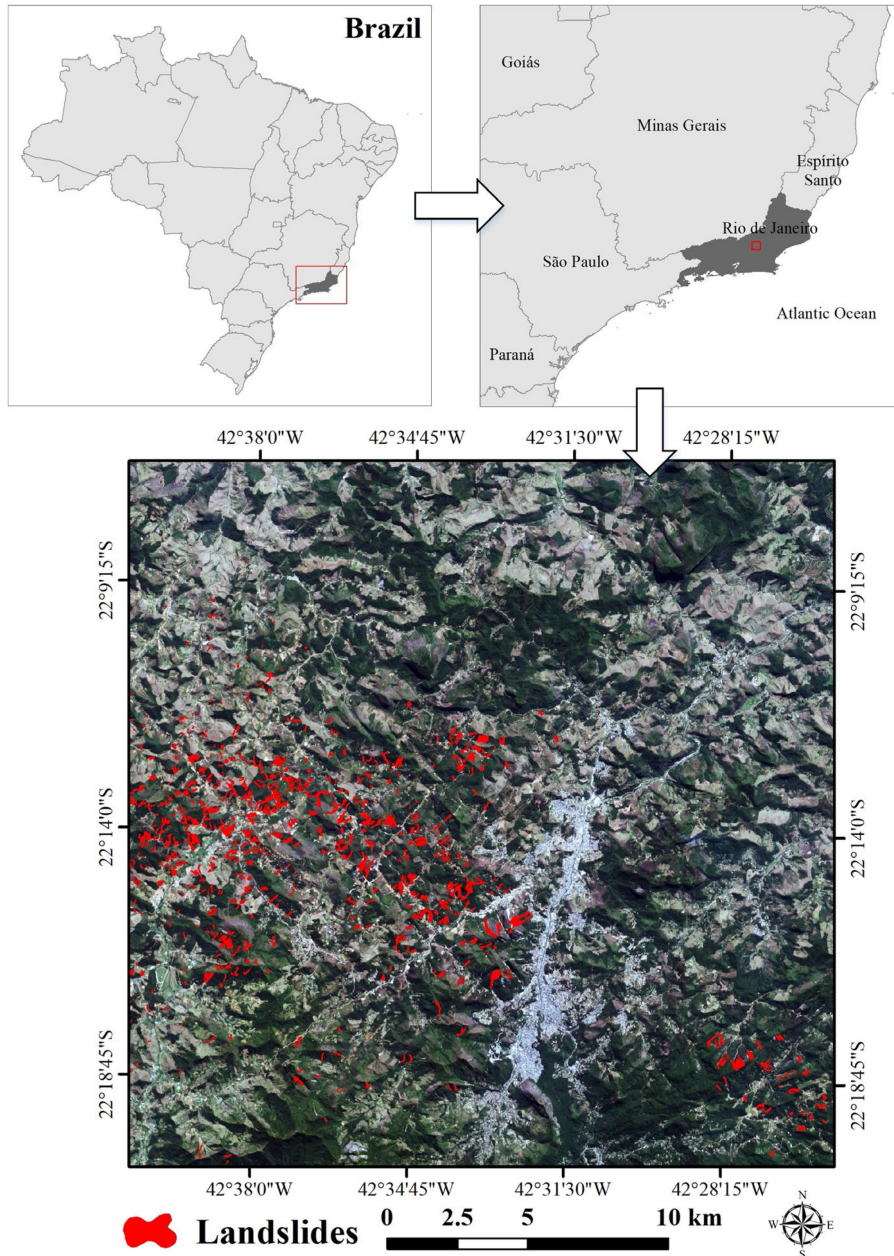
Fig. 1. Location of the study area and annotated rainfall-induced landslides (red polygons).

methods. Section V discusses the effectiveness of the proposed method. Finally, Section VI concludes the article.

II. STUDY AREA AND DATA

The study area is located in the Nova Friburgo mountains in Rio de Janeiro, Brazil, as shown in Fig. 1. In January 2011, a heavy rainfall event of 350 mm/48 h occurred in Nova Friburgo, triggering at least 3500 translational landslides with a total area of about 5.56 km^2, and more than 1500 people died from the rainfall-induced landslides [43]. The landslide inventory map of the study area was manually generated by experts from high-resolution satellite images based on shape and tone information, and validated using Google Earth Pro and compared with landslide maps produced by other scholars [44]. The high-resolution image used in this article was acquired by the RapidEye sensor on August 13, 2011. It has 5000×5000 pixels and five bands in the range of 440–850 nm, with a spatial and temporal resolution of 5 m and 5.5 days, respectively. In the image, 816 landslides caused by heavy rainfall were marked as solid polygons of different sizes, ranging from 200.32 to 78117.35 m^2, as shown in Fig. 1.

III. METHODOLOGY

A. Auxiliary Features

1) NDVI: NDVI, a remote sensing index reflecting the status of land vegetation cover, has been widely employed in land use

TABLE I
CALCULATION FORMULA OF TEXTURE FEATURE FACTOR

Texture feature factor	Formula
Mean	$Mea = \sum_{i,j=0}^{N-1} i \cdot P_{i,j}$
Variance	$Var = \sum_{i,j=0}^{N-1} i \cdot P_{i,j} \left(i - Mea\right)^2$
Homogeneity	$Hom = \sum_{i,j=0}^{N-1} \frac{i \cdot P_{i,j}}{1 + \left(i - j\right)^2}$
Contrast	$Con = \sum_{i,j=0}^{N-1} \left(i - j\right)^2 \cdot P_{i,j}$
Dissimilarity	$Dis = \sum_{i,j=0}^{N-1} i \cdot P_{i,j} \left| i - j \right|$
Entropy	$Ent = \sum_{i,j=0}^{N-1} i \cdot P_{i,j} \ln P_{i,j}$
Second Moment	$Sec = \sum_{i,j=0}^{N-1} i \cdot P_{i,j}{}^2$
Correlation	$Cor = \sum_{i,j=0}^{N-1} i \cdot P_{i,j} \frac{\left(i - Mea\right)\left(j - Mea\right)}{\sqrt{Var_1 \cdot Var_2}}$

and land cover change. In the study of landslide identification, it is generally believed that the vegetation coverage inside the landslide body will change with the movement of the landslide. Therefore, this article introduces the index to strengthen the distinction between vegetation areas and nonvegetation areas

$$NDVI = \frac{NIR - R}{NIR + R} \qquad (1)$$

where NIR and R represent the reflectivity in the near-infrared band and in the red band, respectively.

2) GLCM: The GLCM is a classic analysis method that extracts texture features by calculating the conditional probability density between gray levels of an image, and is widely used for information extraction of remote sensing images. The texture of the landslide is significantly different from the surrounding geological environment. In order to synthesize the information of each band and reduce the dimensionality of data, the principal component analysis is used on the original image, and eight GLCM texture features of the first principal component are calculated. Table I lists the texture features of high-resolution images used for deep learning modeling. The **Mean** indicates how regular the texture distribution is. The stronger the regularity, the greater the mean. **Variance** is the difference between each pixel and the mean. If the gray value of the pixel changes greatly, the variance value is larger. **Homogeneity** is a measure of uniformity within a local area. In this area, the more uniform the image distributed, the greater the homogeneity. **Contrast** represents how the matrix values are distributed and to what extent local variations in the image reflect the sharpness of the image and the depth of the grooves in the texture. The greater the contrast, the deeper the grooves, and the sharper the effect. **Dissimilarity** is similar to **Contrast**. In a local area, the higher the contrast, the stronger the dissimilarity. **Entropy** is a measure

of the randomness of the information contained in an image. Entropy is greatest when all means in the GLCM are equal or the pixel values show the greatest randomness. Therefore, the higher the entropy value, the more complex the grayscale distribution of the image. The **second moment** represents the uniformity of the grayscale distribution of the image and texture thickness. When the image texture is fine and the grayscale distribution is uniform, the second angular moment is larger. **Correlation** refers to the similarity of the grayscale of the image in the row or column direction. The texture reflects the local grayscale correlation of the image. The larger the value is, the greater the correlation.

B. Deep Convolutional Neural Networks

1) Fully Convolutional Network: To apply deep learning CNNs to semantic segmentation, a full convolutional network (FCN) was proposed [45], which replaces the last part of the CNN with a convolution layer (a kernel of size 1 × 1), allowing prediction of two-dimensional dense class label maps [46]. The semantic segmentation effect of point-to-point classification is achieved by restoring the feature map to the input size while preserving the spatial structure of the image. It modifies the fully connected layers of deep CNNs (AlexNet, VGG, GoogleNet, etc.) for image classification into convolutional layers, and proposes an end-to-end semantic segmentation model.

2) U-Net: U-Net was proposed to improve the situation where the target edges are not fine enough in the resultant segmentation map [47]. In this network, the number of channels of the feature map is doubled during each downsampling procedure to allow more feature information (e.g., boundaries, colors, shapes, etc.) to propagate among the convolutional layers, while this number is halved during each upsampling procedure. The skip connection is performed and the stacking operation of dimensional concatenation to fuse feature maps of different scales. This process preserves more high-resolution details to help restore the spatial resolution of the output map and improve segmentation accuracy [48]. This network has been widely used in target extraction from remote sensing images [49]–[51].

3) DeepLabv3+: DeepLabv3+ was proposed to use atrous convolutions to improve segmentation performance [52]. Specifically, it uses atrous convolutions [53] with different strides [54]. The receptive field is expanded through atrous convolution without increasing the network training parameters and reducing the resolution of the output feature map. In essence, DeepLabv3+ fuses the shallow features of the encoder with the deep features of the atrous spatial pyramid pooling [55] module to output more accurate semantic segmentation results. Shallow features can provide better object details, and deep features can provide more abstract semantic information and location information of objects. The fusion of the two features can output higher precision semantic segmentation results.

To better compare the aforementioned methods, Table II lists their advantages and disadvantages.

C. Feature Selection Based on Relief-F

Relief-F is a multiclass feature selection algorithm based on mathematical statistics, calculating the weights of each

TABLE II
COMPARISON OF CHARACTERISTICS OF THREE SEMANTIC SEGMENTATION METHODS

| Algorithm	Characteristics	Advantages	Disadvantages
FCN	It replaces the fully connected layers in the CNN model with fully convolutional layer for pixel-level dense prediction.	FCN solves the problems of repeated storage and convolution computation caused by the use of pixel blocks in traditional segmentation networks.	The deconvolution process is coarse and insensitive to image details.
U-Net	It uses skip connections to stitch the feature maps of the encoding network onto the corresponding feature maps of the decoding network.	On the basis of FCN, a decoding network is added to effectively fuse low-level and high-level features to restore the spatial dimension and boundary information of the image.	The network structure is complex, the number of parameters is large, and the object boundary segmentation performance is not good.
DeepLabv3+	It use the ASPP module to process multi-scale contextual content information for optimization.	The receptive field is enlarged, and more feature information can be extracted.	The computational cost is high, some position information of the pixels is lost, and the local consistency of the feature map is affected.
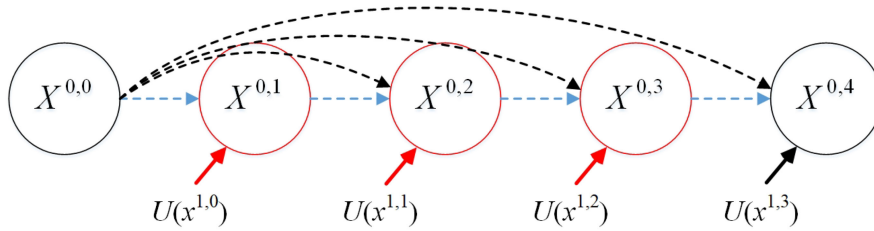
Fig. 2.   Input and output of the first layer of DU-Net.

feature variable and rank the magnitude of the weight values by randomly selecting samples [56]. The Relief-F method can determine the relative importance of the features to landslide occurrence, and it evaluates the value of features by considering the correlation between features and categories [57].

The Relief-F algorithm first selects a random sample R from the dataset D, and then selects k nearest neighbor samples from the similar and different classes of the sample R. In the feature set F, if the distance between samples of different classes is greater than that of samples of similar classes in one feature, the feature is conducive to classification, and its weight is increased. The process repeats n times to take the mean of all the results as the final weight value of each feature. Finally, the feature weight is defined as follows:

$$\omega\left(F_j\right) = \omega\left(F_j\right) - \frac{1}{n \cdot k} \sum_{h \in H} |R_j - h_j| + \frac{1}{n \cdot k} \sum_{m \in M} |R_j - m_j| \tag{2}$$

where $\omega(F_j)$ is the weight value of the jth feature, k is the number of nearest neighbor samples, $\sum_{h \in H} |R_j - h_j|$ indicates the sum of the distances between the selected k similar class neighbor samples and the sample R on the jth feature, and $\sum_{m \in M} |R_j - m_j|$ represents the sum of distances between k heterogeneous nearest neighbor samples and the sample R on the jth feature.

D. Deep U-Net Architecture

DU-Net [58] can effectively combine the advantages of traditional U-Net [47] and DenseNet [59], using skip connection to combine low-level and high-level information in U-Net, while using dense connection to enhance feature propagation in DenseNet. In U-Net, the feature maps of the encoder are received directly in the decoder. However, in DU-Net, the redesigned dense skip connections change the connectivity of the encoder and decoder subnetworks, and the number of convolutional layers depends on the network level. For example, as shown in Fig. 2, the layer-jumping path between nodes $x^{0,0}$ and $x^{0,3}$ is composed of dense convolutional blocks with three convolutional layers. Each convolutional layer is preceded by a connection layer that merges the output of the previous convolutional layer from the same dense block and the corresponding upsampled output of the lower dense block. The DU-Net structure can capture subtle information, obtaining better output than U-Net. Therefore, the use of DU-Net for semantic segmentation of remote sensing images has great application value.

The formula for DU-Net's dense skip connection is given by

$$x^{a,b} = \begin{cases} H\left(x^{a-1,b}\right), & b = 0 \\ H\{[[x^{a,k}]_{k=0}^{b-1}, U\left(x^{a+1,b-1}\right)]\} & b > 0 \end{cases} \tag{3}$$

where $x^{a,b}$ represents the output of the bth node of the ath layer of the network, and $H(\cdot)$, $U(\cdot)$, and $[\cdot]$ represent convolution operation, upsampling operation, and concatenation operation, respectively. When $b = 0$, the node only takes as input from the downsampling layer of the upper layer. There are two inputs passed to the node with $b = 1$, both of which come from the encoding subnetwork and are two consecutive layers. When $b + 1$ inputs are passed to a node with $b > 1$, b of those inputs
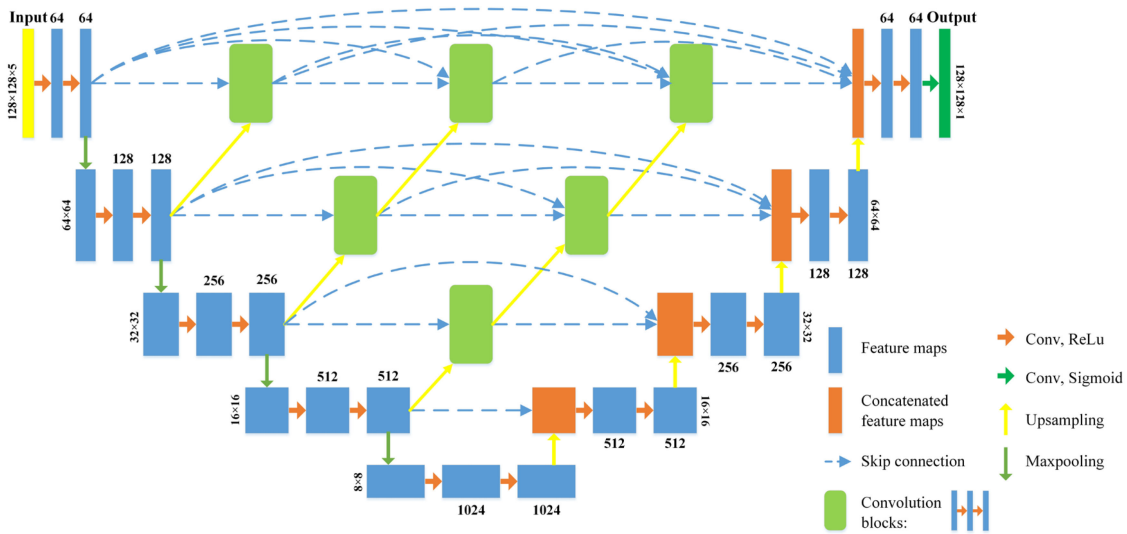
Fig. 3. Architecture of DU-Net.

come from the output of the previous node in the skip path consistent with that node, while one more input is obtained from a skip path lower than that node after an upsampling operation.

We propose an FCDU-Net method to detect landslides, which fuses spectral bands, NDVI, GLCM texture features, and uses the Relief-F algorithm to filter out positive factors from auxiliary features and uses the DU-Net model for feature extraction and training. Fig. 3 shows the DU-Net architecture, consisting of an encoding structure, a decoding structure, and dense skip connections. The feature maps of the same layer have the same size, and from top to bottom, they are the first to fifth layers. According to [58], three, two, and one convolutional modules are included between the first, second, and third layers of the encoder–decoder path, respectively. Each convolution module is composed of two sets of convolutional layers with a kernel size of 3 and a rectified linear unit. The overall feature fusion structure of DU-Net is in the shape of an inverted pyramid. Each layer of the pyramid is equivalent to a dense block, and the features are transferred backward through dense skip connections. Among different layers, the features are deeply aggregated from top to bottom, which shortens the semantic gap when the encoder features and decoder features skip and connect, and improves the feature learning ability of the model.

E. Flowchart of Landslide Inventory Mapping

Fig. 4 illustrates the overall flow of the proposed method. First, training samples are generated through data preprocessing, including image normalization, feature selection, and landslide data processing. The proposed method is supervised and trained based on the training dataset and real samples. Meanwhile, the model is continuously calculated and analyzed based on validation data to ensure satisfactory generalization to unknown datasets. After the validation loss reaches the minimum value, the training process terminates. The model inference process

adopts the edge-ignoring prediction method [60], and the pros and cons of the model are evaluated based on the test dataset.

1) Data Preprocessing: Data preprocessing mainly includes image cropping, normalization, and preparation of landslide training datasets. Different features of the target often have different dimensional units, which will affect the results of data analysis. To eliminate the influence between indicators, image normalization is usually required to make the data indicators in the same order of magnitude and facilitate the training of deep CNNs. In this article, each channel of the original image is normalized by Min–Max normalization so that the input has a similar distribution. Due to the diversity of landslide shapes and sizes, multiple input window sizes are used for landslide detection. First, an n-band high-resolution image and a single-band rasterized ground-truth label layer are stacked to form a $(n+1)$-channel array to ensure that feature transformations remain in the image. We then scan the image using a sliding window algorithm to generate training patches, as shown in Fig. 5.

To better train a deep learning network, it is necessary to select a sufficient number of samples to train the network. If the number of training image sets is small, the scenes contained in these image data may be relatively homogeneous, and the network model cannot learn features extensively, so the generalization ability of the trained model is not strong. Inspired by previous articles in similar fields [61], [62], we use a data augmentation strategy to generate more training samples. Next, these patches are expanded horizontally, vertically, and diagonally, which increases the invariance and robustness of the network to such deformations. All training samples are generated from the training area, and images from the test area are not used during training.

2) Model Training: Using a small-batch training strategy, only a small portion of the training dataset is used as input in each iteration to avoid local minimizing of training errors and achieve rapid convergence in the parameter optimization process [63]. In addition, the scales of landslides are different. We randomly
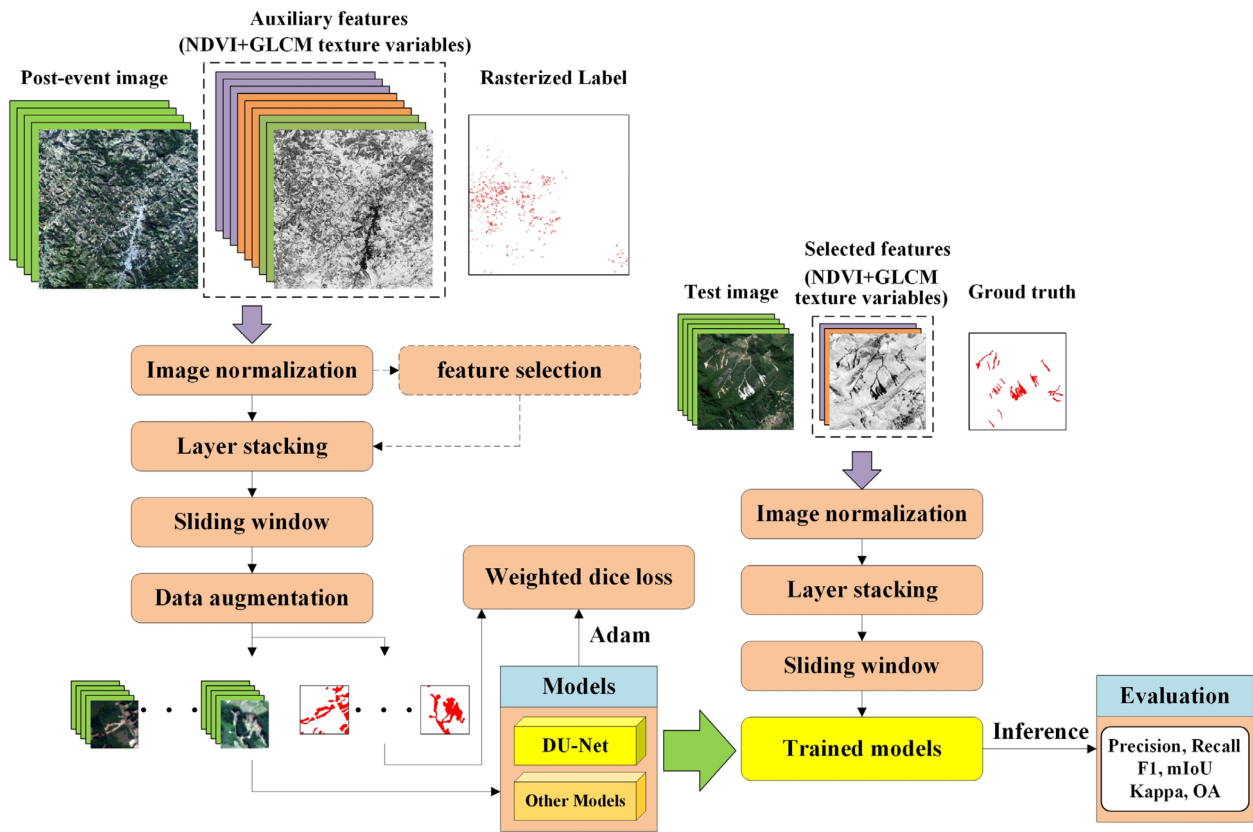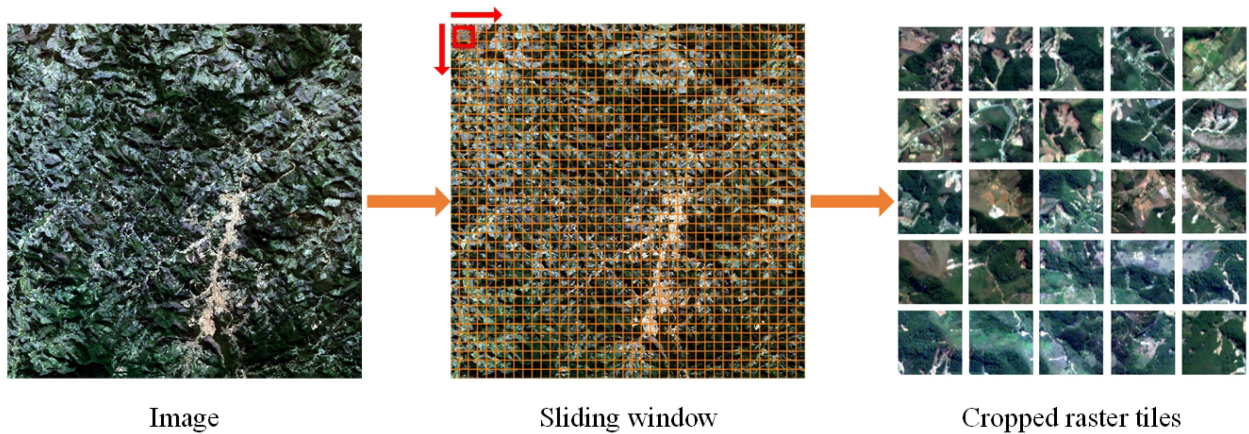
Fig. 4. Flowchart of this article.

Fig. 5. Flowchart of cropping a high-resolution image to generate raster tiles through scanning the image using a sliding window.

selected three test areas. The total training area contains 749 landslides (88%) and test area contains 98 landslides (12%). For imbalanced datasets, this may result in the minority categories being easily ignored since classifiers are usually designed to optimize overall accuracy (OA) [64], [65]. Dice loss [66] is especially proposed as a loss function for medical images with too large proportion of background information and too small proportion of foreground information, resulting in imbalance of positive and negative sample categories. Taking into account the difference in landslide and background proportions, this article uses the weighted dice loss. Specifically, we weight the losses

for different categories and calculate the weight for each class. The higher the frequency, the smaller the lower of the category.

3) Model Validation and Inference: During training, 30% of each dataset is used as validation data, and they are not augmented to evaluate the generalization ability of DCNNs. In this article, a threshold of 0.5 is chosen to classify the results into binary classes. Specifically, if the predicted probability value of a certain pixel is greater than 0.5, the pixel is considered to be a landslide unit, and the pixel with a probability value less than 0.5 is a nonlandslide unit. Next, we conduct model testing to analyze the quality of the network. In particular, we select a
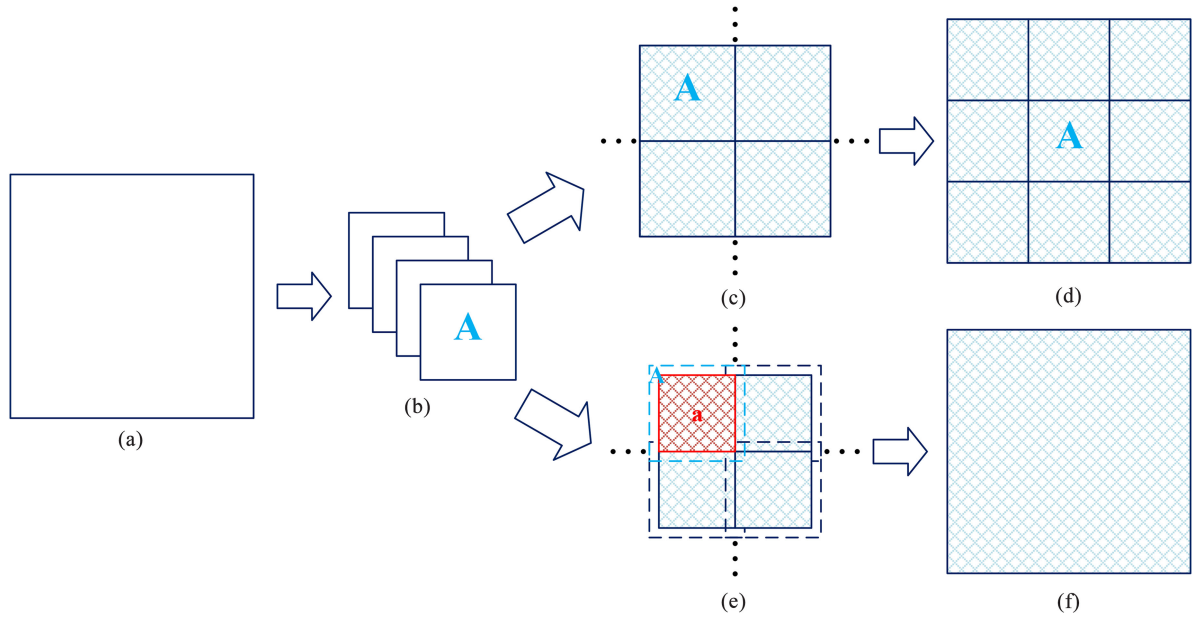
Fig. 6. Sketch map of model inference. (a) Test area image. (b) Cropping the image. (c) Inference results are directly spliced together. (d) Direct splicing results. (e) Splicing inference results while ignoring the edges. (f) Edge-ignoring inference result.

robust model that does not overfit in the training dataset, and evaluate it qualitative and quantitative through visual inspection and different objective metrics.

What is more, the automatic landslide detection model uses datasets from the test area for evaluation. The test area image is sliced into a series of smaller images to be input to the network for inference, and the prediction results are merged into a mosaic to fully cover the original range, as shown in Fig. 6. Direct stitching prediction methods often result in splicing traces, affecting landslide inventory mapping. In this article, we perform edge-ignoring prediction, i.e., cropping overlapping images and adopt the edge-ignoring strategy when splicing. In Fig. 6(e) and (f), the actual cropped image is predicted to be A, then the stitching result is a. The percentage of a in area A is r, and the overlap ratio of adjacent cropped images is $1 - \sqrt{r}$, where the value of r is 0.5.

F. Accuracy Evaluation

To quantitatively analyze the effectiveness of the proposed method, the experiments calculate several evaluation measures of Precision, Recall, and F1 score (F1). Precision indicates that the correctly recognized landslides are divided by the aggregate number of landslides recognized by the model. Recall means that the correctly recognized landslides are divided by the aggregate number of actual landslides. F1 is used as a standard measure of model performance, which represents the harmonic average of the measures of Precision and Recall. Besides, two statistical measures of Kappa and OA are applied to evaluate the predictive ability of different landslide inventory mapping methods. The aforementioned measures are given by

$$\text{Precision} = \frac{T_P}{T_P + F_P} \tag{4}$$

$$\text{Recall} = \frac{T_P}{T_P + F_N} \tag{5}$$

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{6}$$

$$\text{Kappa} = \frac{P_c - P_{\exp}}{1 - P_{\exp}} \tag{7}$$

$$P_{\exp} = \frac{(T_P + F_N)(T_P + F_P) + (F_P + T_N)(F_N + T_N)}{(T_P + T_N + F_N + F_P)^2} \tag{8}$$

$$\text{Overall Accuracy} = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} \tag{9}$$

where T_P and T_N represent the number of positive and negative samples correctly classified by the model, respectively, and F_P and F_N denote the number of positive and negative samples incorrectly classified by the model, respectively. P_c is the observed agreement rate meaning the percentage of type consistent fractions in the two datasets, and $P_{\exp}$ is the expected probability of change agreement. The Kappa coefficient is generally calculated to be between -1 and 1, but usually between 0 and 1. The higher the F1, the better the prediction.

Mean Intersection-over-Union (mIoU) is a commonly adopted validation metric in computer vision. The intersection and union of the predicted area and the true area are obtained, and mIoU is the ratio of these two components. In this article, these two sets are the landslide interpretation map and the landslide prediction map. The value ranges from 0 to 1, where 1 means the best result. The measure of mIoU is given by

$$\text{mIoU} = \frac{1}{m+1} \sum_{j=0}^{m} P_{ij} \frac{P_{ii}}{\sum_{j=0}^{m} P_{ij} + \sum_{j=0}^{m} P_{ji} - P_{ii}} \tag{10}$$
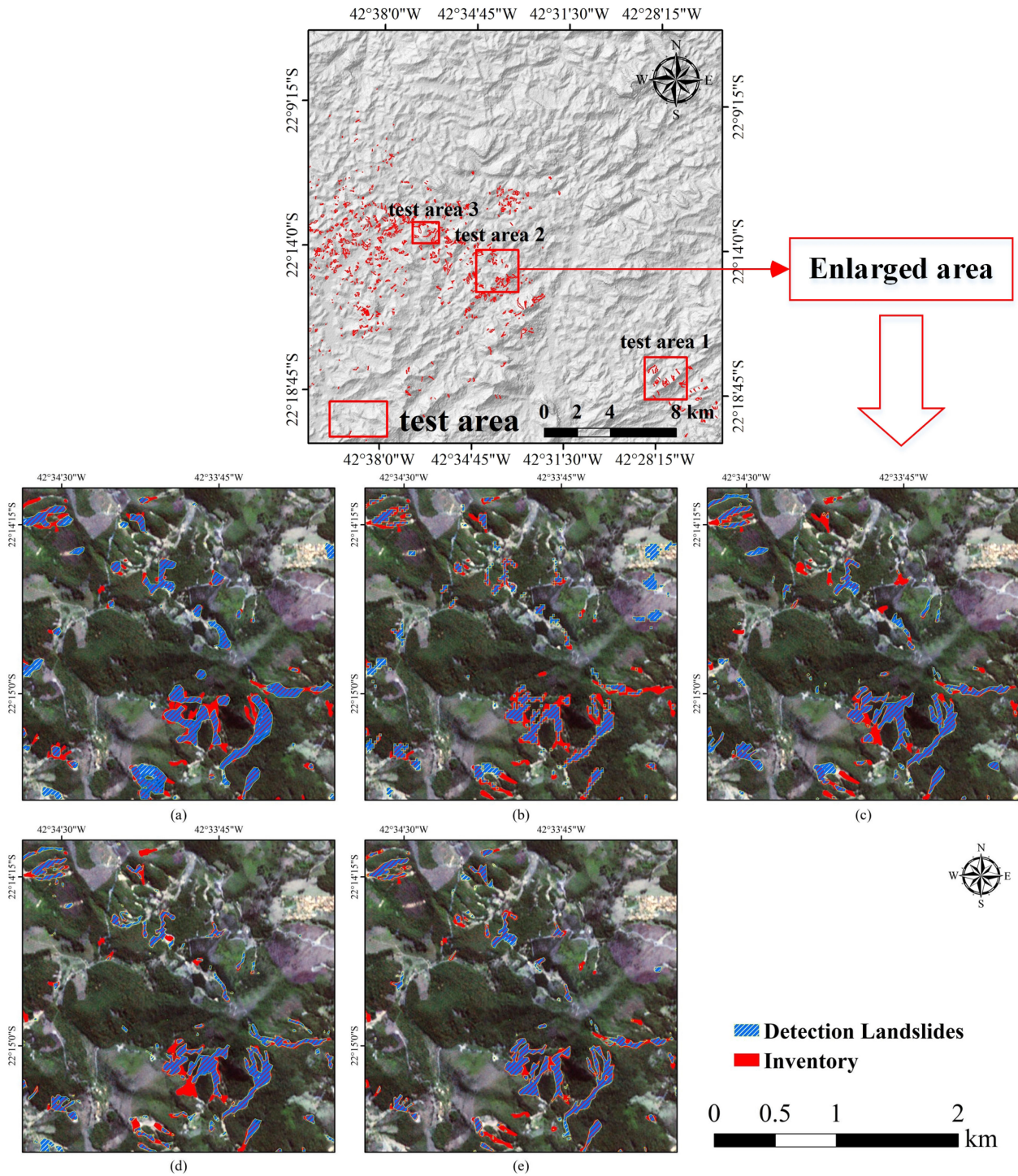
Fig. 7.   Landslide inventory mapping results by different DCNN methods. (a) FCN. (b) DeepLabv3+. (c) U-Net. (d) DU-Net. (e) FCDU-Net.

where $m = 1$ in this article and $m + 1$ is the number of classes. P_{ij} represents the number of real i, but predicted to be j.

IV. RESULTS

A. Experimental Settings

To illustrate the performance of FCDU-Net in landslide inventory mapping, four deep CNNs were compared, including FCN, DeepLabv3+, U-Net, and DU-Net. All deep CNNs were performed on Linux platform using Tensorflow-based Keras framework, and the training and testing process was carried out on a deep learning machine with 64G CPU memory. Calculations were performed using two NVIDIA GeForce RTX 2080Ti GPUs (12 GB RAM) in the NVIDIA CUDA Toolkit 10.1 environment. Experiments all restarted training the network without involving any pretrained model weights. The commonly
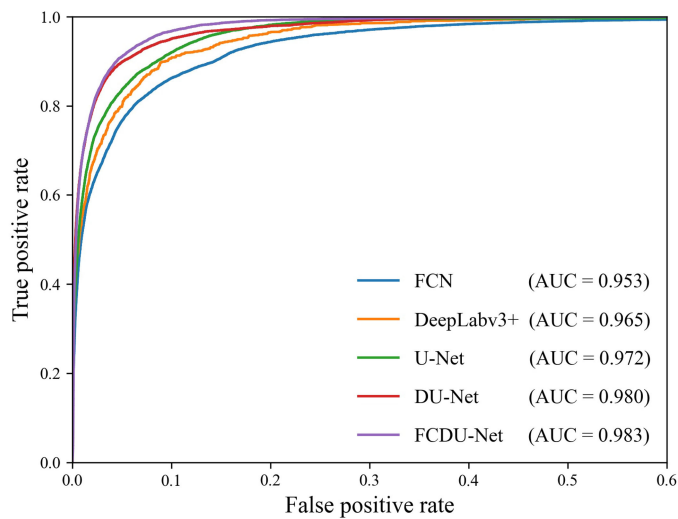
Fig. 8. ROC curves of different methods using the test set.

used Adam optimizer [67] was chosen to update the hyperparameters in the network. Through extensive experiments, 0.001 was used as the learning rate. To avoid overfitting, an early-stopping measure is used to save network parameters when the validation loss curve reaches a minimum. During the training phase, a maximum of 200 training epochs were conducted. To balance training time and model inference performance, a batch size of 64 was chosen. All experimental hyperparameters, training data, and other variables were set identically.

B. Landslide Inventory Mapping Results

All the mentioned deep learning methods were trained using the training dataset, and tested on three subareas of the study area. Referring to previous articles and landslide distribution, all methods used an input window size of 128 × 128. Fig. 7 shows landslide detection maps for the five DCNN methods. It is clear from Fig. 7 that FCN and DeepLabv3+ produced more omission errors in the upper left and lower right corners of the study area, respectively. In contrast, the DU-Net model had fewer false positives and false negatives, compared to U-Net. Fig. 7(e) shows that FCDU-Net has a good recognition for small landslides that are indistinguishable to the naked eye, degree. The landslide boundaries inferred by FCDU-Net match the true distribution of landslides better than U-Net and DU-Net. The visual inspection results show that FCDU-Net has the best prediction performance.

Fig. 8 plots receiver operating characteristic curves of the five models. The area under the curve (AUC) is a quantitative measure of the accuracy of each method in distinguishing categories. In this article, the two categories are landslide areas and nonlandslide areas. Theoretically, the minimum value of AUC is 0 and the maximum is 1. Moreover, the closer the AUC value is to 1, the better the model's ability to identify samples of different classes. As can be seen from Fig. 8, the AUC values of all the models are above 0.95, indicating that DCNN can exhibit very satisfactory predictive ability. Furthermore, the FCDU-Net

method outperforms the other methods with the highest AUC value of 0.983.

Table III lists the detection accuracy evaluations of different methods for Precision, Recall, F1, mIoU, Kappa, and OA, demonstrating the quantitative assessment accuracy of each method based on the test dataset. It can be observed that FCDU-Net has the highest mIoU at 83.19%, which is about 7% higher than U-Net (76.47%). In addition, FCDU-Net achieves the highest Precision, Recall, F1, Kappa, and OA values. Except for Recall, the FCN model performs worse than the other three models in terms of all other statistical metrics.

C. Parametric Analysis

1) Influence of Convolutional Input Window Size: To analyze the influence of convolutional input window size, we conducted landslide detection experiments using FCDU-Net with different convolutional input window sizes. Specifically, the size of 128 × 128 was used as the large input window size of the model, while the sizes of 64 × 64 and 96 × 96 were considered as two different versions of the small window. On this basis, 3060, 1880, and 1296 training samples of size 64 × 64, 96 × 96, and 128 × 128 were derived, respectively. Fig. 9 shows landslide detection maps for two landslide subareas in the study area using FCDU-Net with three different input window sizes. It can be seen that different convolutional input window sizes can reflect the geological environmental conditions of different scales of landslides. As a result, the model can extract different spatial contextual information during training.

To quantitatively analyze the usability of the FCDU-Net method with three different input window sizes, a confusion matrix is used to calculate the four evaluation measures mentioned in Section III-B, as listed in Table IV. The size from 64 × 64 to 96 × 96 improves Precision, but results in lower Recall, mainly because the larger input window has a negative impact on the classification of random points distributed in the landslide polygon. Some randomly distributed points will be close to the boundary of the landslide area. As a result, the nonlandslide area will increase with the size of the input window. However, the lowest F1 and Kappa were obtained based on the size of 96 × 96. In short, FCDU-Net with 128 × 128 can perform better on all evaluation metrics compared to other sizes.

2) Influence of the Relief-F Feature Selection: To verify the effectiveness of DU-Net integrated with Relief-F, the feature importance (FI) of each auxiliary feature is shown in Fig. 10. The higher the FI value, the more important the conditioning factor is to modeling. As can be observed from Fig. 10, the NDVI factor reached the highest FI value. The correlation with landslide occurrence raises as the value increases. Therefore, the conditioning factors that obtained the first three FI values were selected for subsequent modeling.

In this section, we compare the landslide detection performance of DU-Net and FCDU-Net. All experiments used only an input window size of 128 × 128. The test area covers a variety of land use types such as roads, shrubs, woodlands, and houses, and soil landslides mainly occur in woodlands and grasslands. Each landslide varies in shape and size, and some landslides have
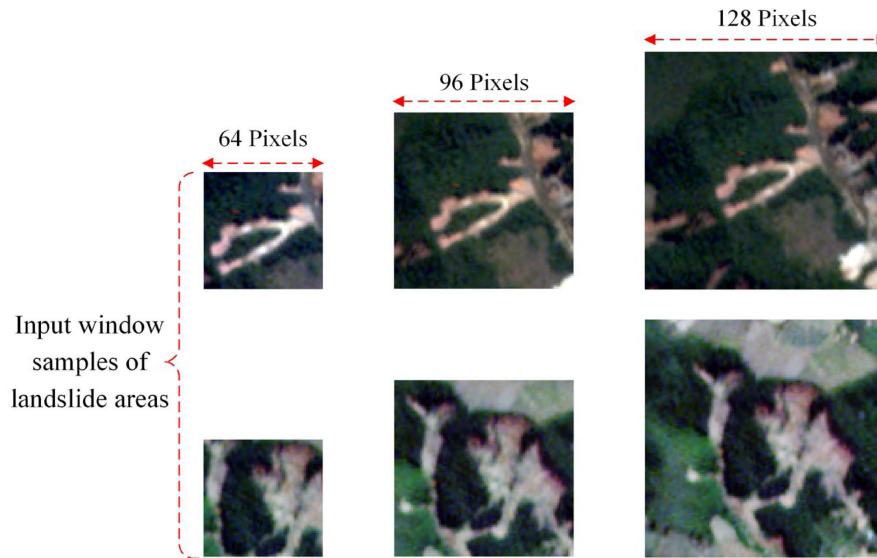
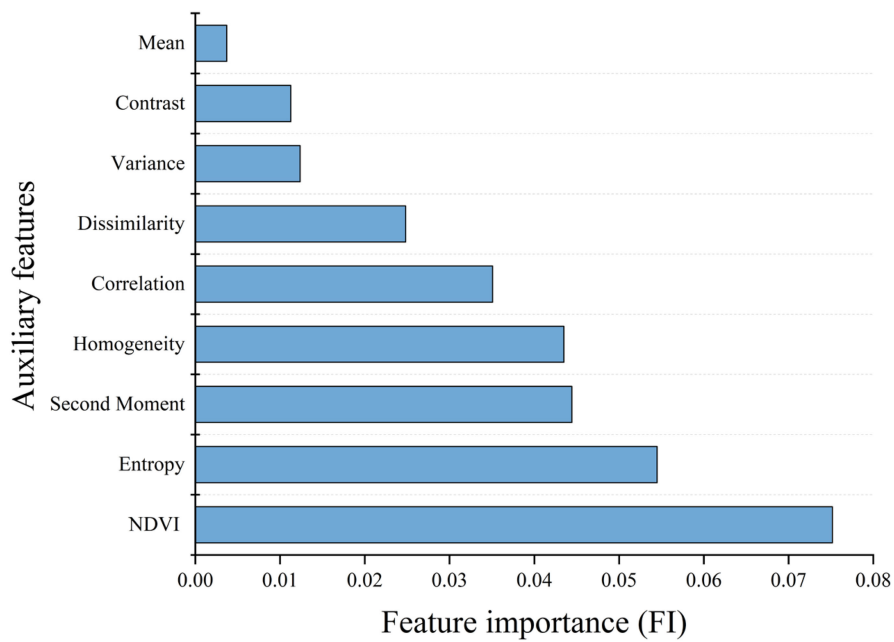Fig. 9. Three convolution input window sizes from two landslide subareas.

Fig. 10. Feature importance of each auxiliary feature factor using Relief-F.

TABLE III
DETECTION ACCURACY ASSESSMENTS BY DIFFERENT METHODS IN ALL TEST AREAS

Methods	mIoU	Precision	Recall	F1	Kappa	OA
FCN	69.41%	54.92%	68.07%	60.79%	58.34%	95.33%
DeepLabv3+	72.23%	68.96%	61.39%	64.96%	63.11%	96.47%
U-Net	76.47%	72.41%	71.24%	71.82%	70.25%	97.02%
DU-Net	78.27%	76.63%	72.38%	74.44%	73.05%	97.35%
FCDU-Net	**83.19%**	**88.87%**	**74.70%**	**81.17%**	**80.21%**	**98.15%**

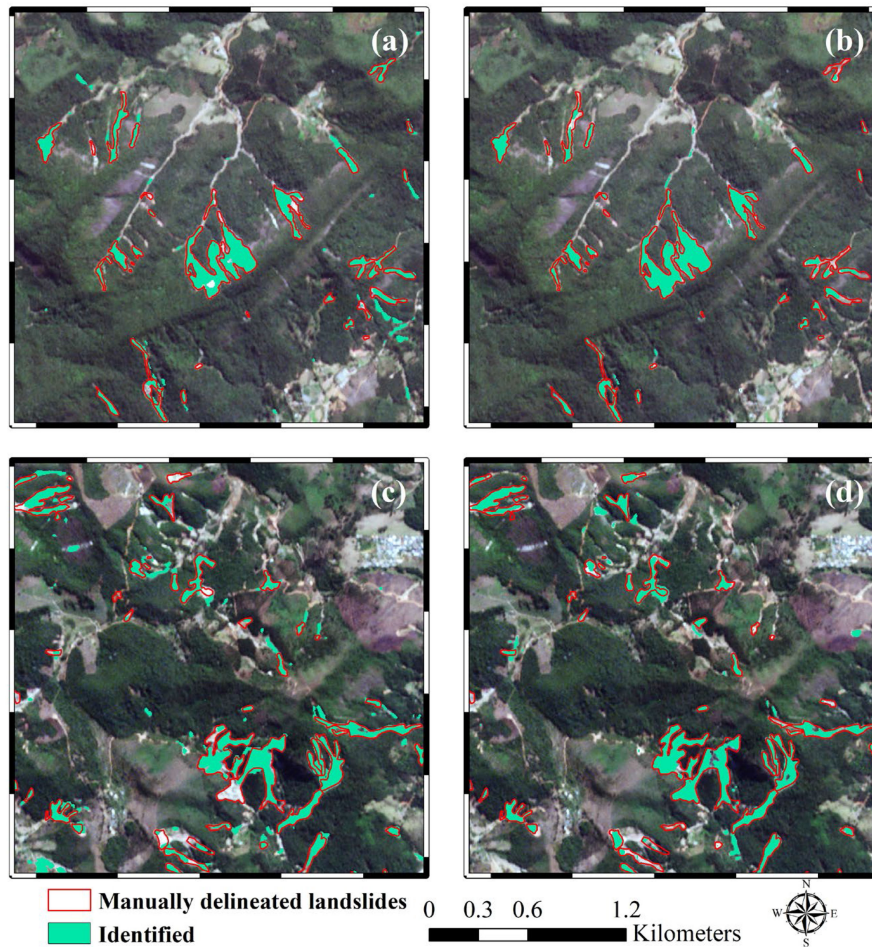The highest accuracies are indicated in bold in each statistical measure.

Fig. 11. Landslide detection maps of two test areas obtained by DU-Net and FCDU-Net. (a) DU-Net on test area 1. (b) FCDU-Net on test area 1. (c) DU-Net on test area 2. (d) FCDU-Net on test area 2.
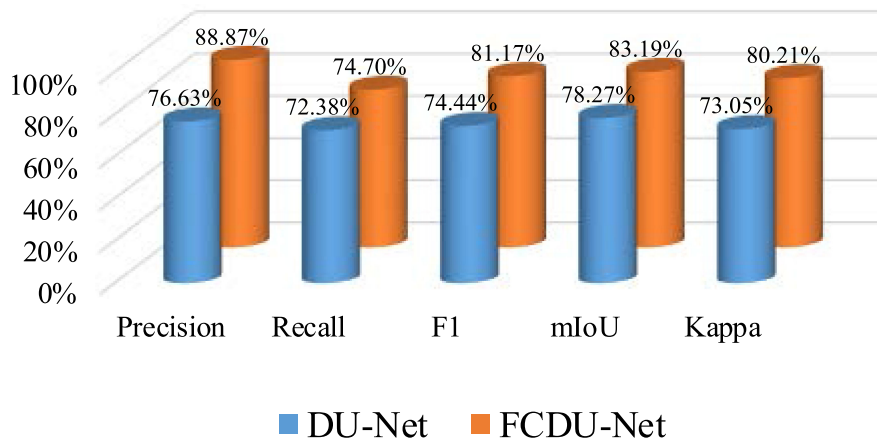

Fig. 12. Performance of DU-Net and FCDU-Net in terms of statistical measures.

spectral signatures similar to bare soil, posing a great challenge to landslide detection. Fig. 11 shows the landslide inventory mapping results for DU-Net and FCDU-Net. It can be seen from Fig. 11(a) and (c) that most of the large landslides can be distinguished from the background by DU-Net, and the boundaries of some slender landslides can also be accurately identified.

However, the results reveal some incomplete and overdetection of small-scale landslides for two reasons. One is the serious imbalance between landslide and nonlandslide samples, and the other is due to the impact of imaging techniques or vegetation coverage, some small landslides cannot be detected by remote sensing images alone. From Fig. 11(b) and (d), we can observe
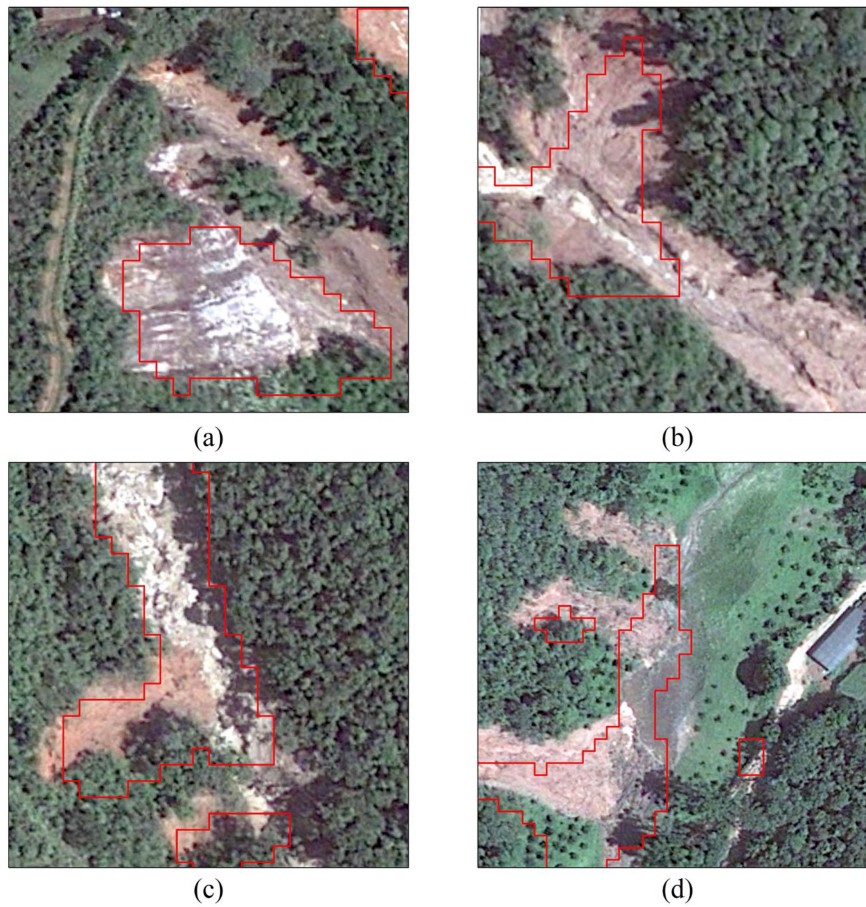
Fig. 13. Very-high-resolution images from Google Earth overlapped with the landslide detection results by FCDU-Net. Figures (a)–(d) denote landslides that were not recorded in the previous landslide inventory data.

TABLE IV
PERFORMANCE OF FCDU-NET USING DIFFERENT INPUT WINDOW SIZES

Sizes	Recall	Precision	F1	Kappa
64×64	73.32%	71.94%	72.62%	71.07%
96×96	65.79%	77.90%	71.34%	69.87%
128×128	74.70%	88.87%	81.17%	80.21%

that the entire landslide map in both figures is satisfactory, reducing the number of overidentified and incorrectly identified landslides, and most of the landslide scars are consistent with the boundaries of the true inventory data. shows the performance of DU-Net and FCDU-Net in terms of five statistical measures Fig. 12previously mentioned in Section III-B. As can be seen from this figure, FCDU-Net performs more balanced in all test domains, and FCDU-Net is better than DU-Net in terms of the statistical measures. Specifically, the Precision of FCDU-Net is 88.87%, which is more than 12% higher than that of DU-Net (76.63%), and the F1, mIoU, and Kappa of FCDU-Net are all 5% higher than that of DU-Net.

V. DISCUSSION

Landslide inventory mapping is the key to emergency rescue and landslide disaster loss assessment [68]. Meanwhile, improving the efficiency of spatial prediction is also important for technical experts to obtain detailed landslide disaster distribution. Therefore, this article compares several DCNNs such as FCN, DeepLabv3+, U-Net, and DU-Net for landslide detection. Furthermore, we evaluate the impact of using different input window sizes on prediction accuracy and ensemble performance of DU-Net and Relief-F.

Generally speaking, high-resolution images have fewer available spectral bands, the spectral distribution of objects in the image varies greatly, and the spectra of different objects overlap each other. However, the geometry and textural information of the object displayed in the image are more obvious. Landslides are usually covered with vegetation. Therefore, it is necessary to combine spectral features with other auxiliary features for landslide detection. Based on the results by Relief-F, NDVI, texture features of Entropy, and Second Moment are the eigenfactors because they represent the characteristic properties and structural information of the landslide. The experimental results show that FCDU-Net achieved higher prediction accuracy than

DU-Net alone because the filtering factor of Relief-F has a positive impact on the detection model and reduces the prediction bias of the model.

To further improve the effectiveness of the proposed method, we apply FCDU-Net to other high-resolution images to identify potential or unlabeled landslide hazards. As shown in Fig. 13, FCDU-Net can manually infer unlabeled landslides in different test areas, which further proves that it can perform well in landslide detection. Furthermore, DCNNs play an active role in updating or supplementing existing landslide databases as they can detect areas that were previously neglected.

In previous articles, some scholars began to explore the application of deep learning or transfer learning in landslide inventory mapping [69]–[74]. A comprehensive quantitative comparison is not conducted herein due to differences in image size and geographic location. They used OBIE, change detection, or long time series detection algorithms, and included several topographic factors (e.g., DEM and its derivatives). For example, Lu and Ma [70] combined transfer learning with OBIE to achieve edge extraction for large landslides. Shi *et al.* [72] proposed a new integrated method combining DCNN and change detection for efficient landslide mapping in the Hong Kong area. Ghorbanzadeh *et al.* [74] showed that topographic factors such as elevation, slope, and plane curvature can improve the prediction performance of deep learning networks. In contrast to these articles, we did not use high-resolution DEM and any postprocessing methods. The excellent performance demonstrates the effectiveness and robustness of the proposed method.

Through comparative analysis, the application of DCNN methods has greater advantages. Specifically, they require less manual fine-tuning than traditional methods and can be easily used in new domains that require only a small amount of training data. Moreover, the trained DU-Net will allow continuous processing and segmentation of high-resolution images, driving the development of software for continuous detection of new landslides.

One of the aims of this article is to explore the effect of landslides at different scales on the detection accuracy of deep learning methods, so we have adopted the strategy of different convolutional input window sizes for the proposed FCDU-Net method. The experimental results demonstrate that FCDU-Net with the convolutional input window size of 128 × 128 can perform the best. It can found that multitemporal or long time series data may be required to train the proposed method to well remove the seasonal changes in mountainous areas. However, by integrating NDVI and eight GLCM texture features with spectral bands in the original image, this article proposes a strategy for landslide feature extraction in high-resolution remote sensing images in southern latitudes in winter, and we will investigate the generalizability of this method in the future.

VI. CONCLUSION

In this article, we propose an effective FCDU-Net semantic segmentation method. The FCDU-Net training model has the advantages of fewer parameters and lower error probability, and can be put into use quickly. To validate the effectiveness of the proposed method, we apply it to landslide detection in remote areas of the Novo Fribourg Mountains. In addition, we also analyze the influence of different convolution input window sizes on the landslide detection results. The experimental results demonstrate that FCDU-Net using an input window size of 128 × 128 can produce the best performance and achieve higher prediction results than classical methods of FCN, DeepLabv3+, U-Net, and DU-Net. Therefore, FCDU-Net has greater application potential in the landslide inventory dynamic mapping system, which will expand the landslide database and promote the subsequent studies of landslide susceptibility mapping. In the future, we will increase landslide samples from other rainfall events and continue to train the proposed model to enhance its robustness to other disaster areas.

REFERENCES

[1] O. Ghorbanzadeh, T. Blaschke, K. Gholamnia, S. R. Meena, D. Tiede, and J. Aryal, "Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection," *Remote Sens.*, vol. 11, no. 2, 2019, Art. no. 196.

[2] S. Ji, D. Yu, C. Shen, W. Li, and Q. Xu, "Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks," *Landslides*, vol. 17, no. 6, pp. 1337–1352, 2020.

[3] A. Mohan, A. K. Singh, B. Kumar, and R. Dwivedi, "Review on remote sensing methods for landslide detection using machine and deep learning," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 7, 2021, Art. no. e3998.

[4] F. Raspini *et al.*, "Updated landslide inventory of the area between the Furiano and Rosmarino creeks (Sicily, Italy)," *J. Maps*, vol. 12, no. 5, pp. 1010–1019, 2016.

[5] G. Danneels, E. Pirard, and H.-B. Havenith, "Automatic landslide detection from remote sensing images using supervised classification methods," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2007, pp. 3014–3017.

[6] P. Lu, A. Stumpf, N. Kerle, and N. Casagli, "Object-oriented change detection for landslide rapid mapping," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 701–705, Jul. 2011.

[7] J. Dou *et al.*, "An integrated artificial neural network model for the landslide susceptibility assessment of Osado Island, Japan," *Natural Hazards*, vol. 78, no. 3, pp. 1749–1776, 2015.

[8] C. Lira *et al.*, "Automatic detection of Landslide features with remote sensing techniques: Application to Madeira Island," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2011, pp. 1997–2000.

[9] J. E. Nichol, A. Shaker, and M.-S. Wong, "Application of high-resolution stereo satellite images to detailed landslide hazard assessment," *Geomorphology*, vol. 76, no. 1/2, pp. 68–75, 2006.

[10] P. Tarolli, G. Sofia, and G. Dalla Fontana, "Geomorphic features extraction from high-resolution topography: Landslide crowns and bank erosion," *Natural Hazards*, vol. 61, no. 1, pp. 65–83, 2012.

[11] Y. Zhang, H. Ran, Y. Peng, and Y. Zheng, "Automatic prediction of landslides over InSAR techniques and differential detection using high-resolution remote sensing images: Application to Jinsha river," in *Proc. China High Resolution Earth Observ. Conf.*, 2019, pp. 471–486.

[12] C. Zhao, Y. Kang, Q. Zhang, Z. Lu, and B. Li, "Landslide identification and monitoring along the Jinsha river catchment (Wudongde Reservoir area), China, using the InSAR method," *Remote Sens.*, vol. 10, no. 7, 2018, Art. no. 993.

[13] E. L. Harp, D. K. Keefer, H. P. Sato, and H. Yagi, "Landslide inventories: The essential part of seismic landslide hazard analyses," *Eng. Geol.*, vol. 122, no. 1/2, pp. 9–21, 2011.

[14] D. Hölbling, B. Friedl, and C. Eisank, "An object-based approach for semi-automated landslide change detection and attribution of changes to landslide classes in northern Taiwan," *Earth Sci. Informat.*, vol. 8, no. 2, pp. 327–335, 2015.

[15] C. Zhong *et al.*, "Landslide mapping with remote sensing: Challenges and opportunities," *Int. J. Remote Sens.*, vol. 41, no. 4, pp. 1555–1581, 2020.

[16] A. Borghuis, K. Chang, and H. Lee, "Comparison between automated and manual mapping of Typhoon-triggered landslides from SPOT-5 imagery," *Int. J. Remote Sens.*, vol. 28, no. 8, pp. 1843–1856, 2007.

[17] E. V. Marcelino, A. R. Formaggio, and E. E. Maeda, "Landslide inventory using image fusion techniques in Brazil," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 11, no. 3, pp. 181–191, 2009.

[18] R. Behling, S. Roessner, H. Kaufmann, and B. Kleinschmit, "Automated spatiotemporal landslide mapping over large areas using rapideye time series data," *Remote Sens.*, vol. 6, no. 9, pp. 8026–8055, 2014.

[19] K. S. Cheng, C. Wei, and S. Chang, "Locating landslides using multi-temporal satellite images," *Adv. Space Res.*, vol. 33, no. 3, pp. 296–301, 2004.

[20] J. Hervás, J. I. Barredo, P. L. Rosin, A. Pasuto, F. Mantovani, and S. Silvano, "Monitoring landslides from optical remotely sensed imagery: The case history of Tessina landslide, Italy," *Geomorphology*, vol. 54, no. 1–2, pp. 63–75, 2003.

[21] J. Nichol and M. S. Wong, "Satellite remote sensing for detailed landslide inventories using change detection and image fusion," *Int. J. Remote Sens.*, vol. 26, no. 9, pp. 1913–1926, 2005.

[22] A. A. Othman and R. Gloaguen, "Automatic extraction and size distribution of landslides in Kurdistan Region, NE Iraq," *Remote Sens.*, vol. 5, no. 5, pp. 2389–2410, 2013.

[23] N. W. Park and K. H. Chi, "Quantitative assessment of landslide susceptibility using high-resolution remote sensing data and a generalized additive model," *Int. J. Remote Sens.*, vol. 29, no. 1, pp. 247–264, 2008.

[24] W. Yang, M. Wang, and P. Shi, "Using MODIS NDVI time series to identify geographic patterns of landslides in vegetated regions," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 4, pp. 707–710, Jul. 2012.

[25] W. Zhang, J. Lin, J. Peng, and Q. Lu, "Estimating Wenchuan earthquake induced landslides based on remote sensing," *Int. J. Remote Sens.*, vol. 31, no. 13, pp. 3495–3508, 2010.

[26] T. R. Martha, N. Kerle, V. Jetten, C. J. van Westen, and K. V. Kumar, "Characterising spectral, spatial and morphometric properties of landslides for semi-automatic detection using object-oriented methods," *Geomorphology*, vol. 116, no. 1/2, pp. 24–36, 2010.

[27] T. R. Martha, N. Kerle, C. J. Van Westen, V. Jetten, and K. V. Kumar, "Object-oriented analysis of multi-temporal panchromatic images for creation of historical landslide inventories," *ISPRS J. Photogramm. Remote Sens.*, vol. 67, pp. 105–119, 2012.

[28] A. Stumpf and N. Kerle, "Object-oriented mapping of landslides using random forests," *Remote Sens. Environ.*, vol. 115, no. 10, pp. 2564–2577, 2011.

[29] Y. Martin and S. Franklin, "Classification of soil-and bedrock-dominated landslides in British Columbia using segmentation of satellite imagery and DEM data," *Int. J. Remote Sens.*, vol. 26, no. 7, pp. 1505–1509, 2005.

[30] J.-Y. Rau, J.-P. Jhan, and R.-J. Rau, "Semiautomatic object-oriented landslide recognition scheme from multisensor optical imagery and DEM," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 2, pp. 1336–1349, Feb. 2013.

[31] S. Heleno, M. Matias, P. Pina, and A. J. Sousa, "Semiautomated object-based classification of rain-induced landslides with VHR multispectral images on Madeira Island," *Natural Hazards Earth Syst. Sci.*, vol. 16, no. 4, pp. 1035–1048, 2016.

[32] F. Chen, B. Yu, and B. Li, "A practical trial of landslide detection from single-temporal Landsat8 images using contour-based proposals and random forest: A case study of National Nepal," *Landslides*, vol. 15, no. 3, pp. 453–464, 2018.

[33] Z. Li, W. Shi, S. W. Myint, P. Lu, and Q. Wang, "Semi-automated landslide inventory mapping from bitemporal aerial photographs using change detection and level set method," *Remote Sens. Environ.*, vol. 175, pp. 215–230, 2016.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[35] X. X. Zhu *et al.*, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[36] D. He, Q. Shi, X. Liu, Y. Zhong, and X. Zhang, "Deep subpixel mapping based on semantic information modulated network for urban land use mapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10628–10646, Dec. 2021.

[37] H. Guo, Q. Shi, A. Marinoni, B. Du, and L. Zhang, "Deep building footprint update network: A semi-supervised method for updating existing building footprint from bi-temporal remote sensing images," *Remote Sens. Environ.*, vol. 264, 2021, Art. no. 112589.

[38] D. He, Y. Zhong, X. Wang, and L. Zhang, "Deep convolutional neural network framework for subpixel mapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9518–9539, Nov. 2020.

[39] A. Ding, Q. Zhang, X. Zhou, and B. Dai, "Automatic recognition of landslide based on CNN and texture change detection," in *Proc. 31st Youth Acad. Annu. Conf. Chin. Assoc. Automat.*, 2016, pp. 444–448.

[40] H. Yu, Y. Ma, L. Wang, Y. Zhai, and X. Wang, "A landslide intelligent detection method based on CNN and RSG_R," in *Proc. IEEE Int. Conf. Mechatron. Automat.*, 2017, pp. 40–44.

[41] Y. Yi and W. Zhang, "A new deep-learning-based approach for earthquake-triggered landslide detection from single-temporal Rapideye satellite imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 6166–6176, Oct. 2020.

[42] S. Qin *et al.*, "Landslide detection from open satellite imagery using distant domain transfer learning," *Remote Sens.*, vol. 13, no. 17, 2021, Art. no. 3383.

[43] A. S. Avelar, A. L. C. Netto, W. A. Lacerda, L. B. Becker, and M. B. Mendonça, "Mechanisms of the recent catastrophic landslides in the mountainous range of Rio de Janeiro, Brazil," in *Landslide Science and Practice*. Berlin, Germany: Springer, 2013, pp. 265–270.

[44] A. L. C. Netto *et al.*, "January 2011: The extreme landslide disaster in Brazil," in *Landslide Science and Practice*. Berlin, Germany: Springer, 2013, pp. 377–384.

[45] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[46] W. Sun and R. Wang, "Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with DSM," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 3, pp. 474–478, Mar. 2018.

[47] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.

[48] V. Iglovikov and A. Shvets, "Ternausnet: U-net with vgg11 encoder pretrained on imagenet for image segmentation," 2018, *arXiv:1801.05746*.

[49] M. Volpi and D. Tuia, "Dense semantic labeling of subdecimeter resolution images with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 881–893, Feb. 2016.

[50] X. Wei *et al.*, "Semantic pixel labelling in remote sensing images using a deep convolutional encoder–decoder model," *Remote Sens. Lett.*, vol. 9, no. 3, pp. 199–208, 2018.

[51] L. P. Soares, H. C. Dias, and C. H. Grohmann, "Landslide segmentation with u-net: Evaluating different sampling methods and patch sizes," 2020, *arXiv:2007.06672*.

[52] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.

[53] S. Mallat, *A Wavelet Tour of Signal Processing*. Amsterdam, The Netherlands: Elsevier, 1999.

[54] A. R. Choudhury, R. Vanguri, S. R. Jambawalikar, and P. Kumar, "Segmentation of brain tumors using deeplabv3+," in *Proc. Int. MICCAI Brainlesion Workshop*, 2018, pp. 154–167.

[55] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[56] X. Meng, P. Zhang, and D. Zhang, "Decision tree for online voltage stability margin assessment using C4.5 and Relief-F algorithms," *Energies*, vol. 13, no. 15, 2020, Art. no. 3824.

[57] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Proc. Mach. Learn.*, 1992, pp. 249–256.

[58] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Berlin, Germany: Springer, 2018, pp. 3–11.

[59] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[60] Z. Wang, Y. Zhou, S. Wang, F. Wang, and Z. Xu, "House building extraction from high resolution remote sensing image based on IEU-Net," *J. Remote Sens.*, vol. 10, 2021, Art. no. 1768.

[61] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[62] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[63] J. Konečný, J. Liu, P. Richtárik, and M. Takáč, "Mini-batch semi-stochastic gradient descent in the proximal setting," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 2, pp. 242–255, Mar. 2015.

[64] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu, and J. Li, "Dice loss for data-imbalanced NLP tasks," 2019, *arXiv:1911.02855*.

[65] S. Visa and A. Ralescu, "Issues in mining imbalanced data sets-a review paper," in *Proc. 16th Midwest Artif. Intell. Cogn. Sci. Conf.*, 2005, pp. 67–73.

[66] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis.*, 2016, pp. 565–571.

[67] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[68] P. Zhang, C. Xu, S. Ma, X. Shao, Y. Tian, and B. Wen, "Automatic extraction of seismic landslides in large areas with complex environments based on deep learning: An example of the 2018 Iburi Earthquake, Japan," *Remote Sens.*, vol. 12, no. 23, 2020, Art. no. 3992.

[69] W. Qi, M. Wei, W. Yang, C. Xu, and C. Ma, "Automatic mapping of landslides by the ResU-Net," *Remote Sens.*, vol. 12, no. 15, 2020, Art. no. 2487.

[70] H. Lu *et al.*, "Landslides information extraction using object-oriented image analysis paradigm based on deep learning and transfer learning," *Remote Sens.*, vol. 12, no. 5, 2020, Art. no. 752.

[71] L. Bragagnolo, L. R. Rezende, R. V. da Silva, and J. M. V. Grzybowski, "Convolutional neural networks applied to semantic segmentation of landslide scars," *Catena*, vol. 201, 2021, Art. no. 105189.

[72] W. Shi, M. Zhang, H. Ke, X. Fang, Z. Zhan, and S. Chen, "Landslide recognition by deep convolutional neural network and change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 4654–4672, Jun. 2021.

[73] P. Liu *et al.*, "A research on landslides automatic extraction model based on the improved mask R-CNN," *ISPRS Int. J. Geo-Inf.*, vol. 10, no. 3, 2021, Art. no. 168.

[74] O. Ghorbanzadeh, S. R. Meena, H. S. S. Abadi, S. Tavakoli Piralilou, L. Zhiyong, and T. Blaschke, "Landslide mapping using two main deep-learning convolution neural network (CNN) streams combined by the Dempster–Shafer (DS) model," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 452–463, Dec. 2020.

Guosen Xu received the B.E. degree in resource exploration engineering from Hebei GEO University, Shijiazhuang, China, in 2019. He is currently working toward the M.S. degree in earth exploration and information technology at the China University of Geosciences, Wuhan, China.

His research interests include landslide detection, deep learning, and remote sensing applications.

Yi Wang (Member, IEEE) received the B.S. degree in printing engineering and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2002 and 2007, respectively.

He is currently a Professor with the Institute of Geophysics and Geomatics, China University of Geosciences (CUG), Wuhan, China. He is the Head of the Department of Geoinformatics. His research interests include remote sensing technology and application, geological and ecological environment monitoring, and data intelligent processing.

Prof. Wang is a member of Geological Society of China and Chinese Association of Automation. In 2019, he was named CUG Outstanding Young Talent.

Lizhe Wang (Fellow, IEEE) received the B.E. and M.E. degrees in electrical engineering and automation from Tsinghua University, Beijing, China, in 1998 and 2001, respectively, and the D.E. degree in applied computer science from the University of Karlsruhe, Karlsruhe, Germany, in 2007.

He is currently the Dean of the School of Computer Science, China University of Geosciences, Wuhan, China. His research interests include remote sensing data processing, digital earth, and big data computing.

Dr. Wang is a Fellow of SPIE, IET, and BCS, and an Associate Editor for the *Remote Sensing* and *International Journal of Digital Earth*.

Lucas Pedrosa Soares received the B.S. degree in geology from the Institute of Geosciences, University of Brasília, Brasília, Brazil, in 2018. He is currently working toward the master's degree in remote sensing applications with deep learning with the Institute of Geosciences of the University of São Paulo, São Paulo, Brazil.

He is also a Researcher with Spatial Analysis and Modelling Lab. His research interests include landslide detection, landslide susceptibility mapping, artificial intelligence, deep and machine learning, and remote sensing applications with deep learning.

Carlos H. Grohmann received the B.S. degree in geology, and the M.Sc. and Ph.D. degrees in geotectonics from the University of Sao Paulo, Brazil, in 2002, 2004, and 2008, respectively.

In 2007, he was a Visiting Researcher with Kingston University London, Kingston-upon-Thames, U.K., and in 2009, he was employed as a Full-Time Professor with the Institute of Geosciences, University of Sao Paulo. He is currently an Associate Professor of geotechnologies with the Institute of Energy and Environment of the University of Sao Paulo, where he runs the Spatial Analysis and Modelling Lab. His research interests include digital terrain analysis (geomorphometry), remote sensing, applications of data collected from remotely piloted aircrafts in Earth and environmental sciences, LiDAR, and structure from motion-multiview stereo.

Dr. Grohmann has been a Board Member of the Brazilian Geological Society from 2014 to 2019 and of the International Society for Geomorphometry from 2015 to 2021. He is an Associate Editor for *Computers & Geosciences* and is the present Editor-in-Chief of the *Brazilian Journal of Geology*.