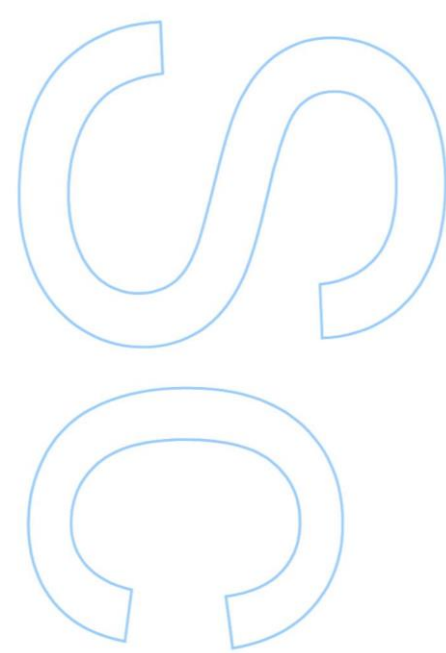
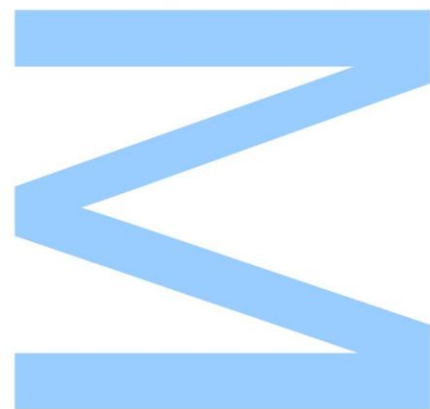


Automatic detection of earthquake's deformations in SAR Interferograms



Bruno Miguel Ferreira da Silva

Master in Remote Sensing

Department of Geoscience, Environment and land use planning

2021

Supervisor

António Cunha, Auxiliar professor, University Trás-os-Montes e Alto Douro

Co-Supervisor

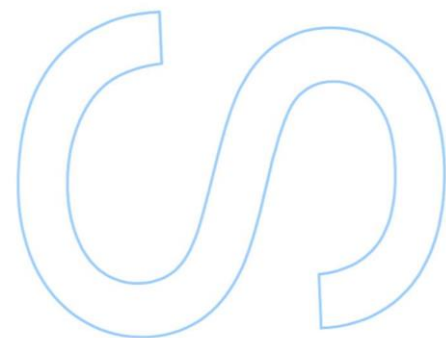
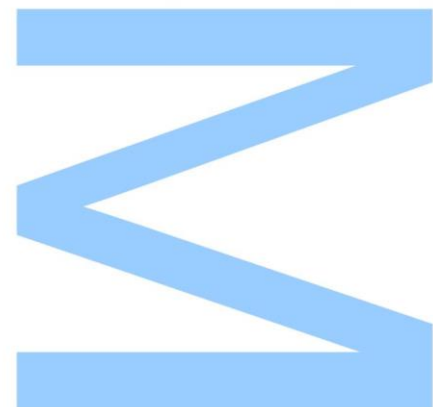
Joana Fernandes, Professor, School of Science from University of Porto



Todas as correções determinadas
pelo júri, e só essas, foram efetuadas.

O Presidente do Júri,

Porto, ____/____/____



Acknowledgements

I want to thank my supervisor António Cunha for all his help and for always being present and available when needed. It did not matter what day and time of the week he was there to help. Thank you for making me trust in myself and for all the available hours to teach me and improve this project. During this time was more than a professor. He was a friend. I am also grateful to my co-supervisor Joana Fernandes for the support. I also want to thank Professor Joaquim João Sousa for the support and for always following me and this project throughout this time, helping in all I needed. Can not forget to show my gratitude all my professors who passed their knowledge to me during these years.

I need to show my gratitude to my friends Luís Costa and Alexandre Neto for all the advice for this work. I can not forget all my friends and colleagues who have been with me all these years. Together we leave unforgettable moments.

Andreia Rocha, thank you for all the support and for encouraging me whenever I was down. You made me mature and someone with goals. I hope to continue to grow with you.

Finally, I want to thank my parents and brothers for be always there. In special to my parents for all the sacrifices they made to make my and my brother's life easier. If we are where we are, it is all thanks to them. I will never forget that.

Resumo

Foi apenas no século XX que os cientistas descobriram que a ocorrência de terremotos se deve à liberação da energia acumulada nas falhas tectônicas. Esses fenômenos naturais ocorrem com diferentes intensidades e escalas. As consequências destes fenômenos naturais, que podem ser sentidos a muitos quilômetros de distância do epicentro, dependem da magnitude e da profundidade a que ocorrem. É, por isso, crucial a delimitação da área afetada por um episódio desta natureza, para evitar riscos após o evento, como o desabamento de edifícios.

Os desenvolvimentos das técnicas de interferometria RADAR de Abertura Sintética (InSAR), na última década do século passado, proporcionaram o cálculo de imagens de deformações terrestres (interferogramas) com precisões centimétricas, permitindo estudar deformações provocadas por deslizamentos de terra, movimento de glaciares, variação dos níveis dos aquíferos e outros perigos naturais e artificiais.

O lançamento da missão Sentinel-1 A/B, pela Agência Espacial Europeia (ESA), possibilitou o acesso a um vasto conjunto de dados SAR à escala global e de forma totalmente gratuita. Fazendo uso desta informação, o “*Centre for Observation and Modelling of Earthquakes, Volcanoes & Tectonics*” (COMET), gerou os interferogramas sobre as regiões mais ativas do planeta e disponibilizou-os de forma totalmente gratuita para serem utilizados por todos os interessados. Tendo em conta que a missão Sentinel-1 gera mais de 10 TB de dados por dia, seria impraticável processar e analisar toda a informação, através de métodos tradicionais-. A Inteligência Artificial, nomeadamente, as técnicas de *Machine Learning*, apresentam-se como a oportunidade para automatizar estes processos de processamento e de análise.

Esta dissertação teve como objetivo principal explorar técnicas de *Machine Learning* e *Deep Learning* para detetar, de forma automática, deformações da crosta terrestre provocadas por eventos sísmicos e identificáveis em interferogramas SAR. A partir dos dados públicos, disponibilizados pelo COMET, foi criado um conjunto de dados com anotações das deformações, para treinar e avaliar os modelos de *Deep Learning*. Os resultados alcançados pelos métodos de detecção foram satisfatórios e corroboram a ideia de que podem ser usados para detetar, com sucesso e de forma automática, deformações sísmicas - o melhor modelo de classificação alcançou uma área sob a curva ROC (AUC) de 0,86. Relativamente aos modelos de segmentação, apresentaram um comportamento menos eficiente, tendo o melhor modelo atingido um coeficiente de *Dice* de 0,64 – este resultado deve-se, provavelmente, ao tamanho

resduzido do conjunto de dados criado. Mais testes terão de ser realizados para testar esta hipótese.

Palavras-chave: InSAR, Franjas de deformação, Aprendizagem Máquina, Aprendizagem Profunda, Ciência de dados

Abstract

In the 20th century, scientists discovered that earthquakes occur due to the release of energy accumulated in tectonic faults. These natural phenomena occur with different intensities and scales. Depending on their magnitude and depth, they can be felt from a few hundred meters to several kilometres from their epicentre and cause significant damage and fatalities. Therefore, knowing the affected area is essential to avoid risks, such as the collapse of buildings in short/medium term after the event.

Since 1992, with the development of Interferometric Synthetic Aperture Radar (InSAR) techniques, it has been possible to calculate images of terrestrial deformations across the globe with precision in the order of centimetres (or less) from Synthetic Aperture Radar (SAR). It allowed the study of landslides, glaciers, aquifers, and other natural and man-made hazards. The European Space Agency (ESA) developed the Sentinel-1 mission and provided free access to the acquired SAR data. The “Centre for Observation and Modeling of Earthquakes, Volcanoes & Tectonics” (COMET) carried out the heavy InSAR calculations and made available the respective interferograms also to free access. This information is very important, but Sentinel-1 alone generates more than 10 TB of data per day, making manual processing unviable. In this context, Artificial Intelligence (AI) techniques present themselves as the opportunity to automate the processing of such data.

This dissertation's main objective aims to explore AI techniques, as Machine Learning and Deep Learning techniques, for automation of seismic deformations detection in SAR interferograms and to estimate the earthquake's impact area. A dataset with deformation annotations was created from COMET public data to train and evaluate Deep Learning models. The results achieved by the detection methods were satisfactory and supported the idea that they can be used to automatically detect seismic deformations successfully - the best classification model achieved an area under curve ROC (AUC) of 0.86. The segmentation models were more limited, with the best model reaching a Dice coefficient of 0.64 – this result is probably due to the small size of the created dataset. More tests need to be carried out in the near future to test this hypothesis.

Keywords: InSAR, Deformation Fringes, Machine Learning, Deep Learning, Data Science

“The first step is to establish that something is possible; then probability will occur.”

Elon Musk

Contents

1. Introduction.....	1
1.1. Motivation.....	1
1.2. Objectives.....	2
1.3. Publications.....	3
1.4. Document structure.....	4
2. Literature review.....	5
2.1. InSAR fundamentals.....	5
2.1.1. Synthetic Aperture Radar.....	5
2.1.2. Interferometric Synthetic Aperture Radar and earthquakes.....	7
2.1.3. Sentinel-1.....	9
2.2. Machine learning.....	11
2.2.1. Artificial Intelligence.....	11
2.2.2. Artificial Neural Networks.....	12
2.2.3. Deep learning.....	16
2.3. Deep learning methods.....	17
2.3.1. Architectures for classification.....	17
2.3.2. Architectures for segmentation.....	19
2.4. Deep Learning for InSAR.....	21
2.4.1. Earthquake deformations detection.....	21
2.4.2. Volcanoes deformations detection.....	21
2.4.3. Slow ground deformation detection.....	22
2.5. Summary.....	23
3. Material and methods.....	25
3.1. Data.....	26
3.2. Data preparation.....	32
3.3. Models train.....	35
3.3.1. Dealing with unbalanced data and small datasets.....	35
3.3.2. Classification models.....	37
3.3.3. Segmentation models.....	39
3.4. Model evaluation.....	39
3.4.1. Classification metrics.....	40
3.4.2. Segmentation metrics.....	42

4. Results and discussion.....	43
4.1. Classification.....	43
4.2. Segmentation	52
5. Conclusion	57
References.....	61
Attachments.....	69

List of Figures

Figure 1: Illustration of a space-borne SAR sensor. Source:(Raab, 2016).....	6
Figure 2: INSAR functioning, source: (Biggs & Wright, 2020)	8
Figure 3: Interferograms from an earthquake in California (2019). Inside the box, it is possible to see the deformation area as the fault represented by the line: (Figures obtained in LICSBAS database).....	8
Figure 4: Sentinel-1 modes	10
Figure 5: Evolution of artificial intelligence.....	11
Figure 6: Machine Learning creates a new way of looking at data. This type of system is trained rather than explicitly programmed (adapted figure from Chollet, 2018).....	11
Figure 7: Representation of a Neural Network (adapted figure from Sharma et al., 2020)	12
Figure 8: Activation functions a) ReLu; b) Sigmoid	13
Figure 9: Usual architecture of a neural network (adapted figure from Chollet, 2018)......	13
Figure 10: Example of the architecture of CNN with max-pooling operation using a mask of 3x3 matrix.	17
Figure 11: Idea behind inception model. Adapted figure from Szegedy et al., 2015	18
Figure 12: Architecture of VGG19	18
Figure 13: A residual block of ResNet and ResNetV2. Adapted image from (Van Hieu & Hien, 2020)	19
Figure 14: Architecture of Unet. Credit: Weng & Zhu, 2015.....	20
Figure 15: Dual attention architecture. credit: Fu et al., 2018.....	20
Figure 16: Proposed methodology pipeline.....	25
Figure 17: Earthquakes zones used in this work.....	27
Figure 18: Earthquakes magnitude distribution for downloaded interferograms	28
Figure 19: Examples of interferograms from big earthquakes zones	30
Figure 20: Examples of interferogram from medium earthquakes zones	31
Figure 21: Examples of interferograms from small earthquakes.....	31
Figure 22: Data preparation pipeline	32
Figure 23: Earthquake interferogram (left) and his mask (right)	32
Figure 24: Demonstration of interferogram rotation	33
Figure 25: Demonstration of patch creation.....	34
Figure 26: Example of data augmentation.....	36
Figure 27: Comparison between cross-entropy and focal loss, adapted figure from T. Lin 2017	36
Figure 28: Architecture of the classification models.....	38
Figure 29: Confusion Matrix.....	40

Figure 30: Example of a ROC Curve	41
Figure 31: ROC Curves for the preliminary tests. At left wrapped interferograms, at right, unwrapped interferograms.....	44
Figure 32: ROC Curves for models using data augmentation on the final dataset. At left wrapped interferograms, at right, unwrapped interferograms.....	45
Figure 33: ROC Curves of the preliminary models applied on the final test set. At left wrapped interferograms, at right, unwrapped interferograms	47
Figure 34: ROC Curves for the classification models using focal loss: at left the ROC curve for wrapped interferograms, and at right the ROC curves for unwrapped interferograms.....	49
Figure 35: ROC Curve, Focal Loss vs. Data augmentation.....	50
Figure 36: Overlapped earthquakes patches and its probability of having deformation earthquake according to the model.....	50
Figure 37: Segmentation confidence area for two earthquakes using a) InceptionV3, b)VGG19, c)ResNet50V2, and d) Dual attention	53
Figure 38: Segmentation borders for two earthquakes using a,b) InceptionV3, c,d)VGG19; - 50%threshold, -best threshold.....	54
Figure 39: Bad segmentation borders for two earthquakes using a) InceptionV3, b)VGG19; - 50%threshold, -best threshold.....	55

List of Tables

Table 1: The various frequency bands.	6
Table 2: Information about selected earthquakes.....	29
Table 3: Number of interferograms for each set of each dataset (wrapped and unwrapped) ..	33
Table 4: Final Dataset in patches (number of interferogram patches for a set).....	35
Table 5: Preliminary dataset (number of patches)	38
Table 6: Metrics for the preliminary tests. The threshold at 50%.	43
Table 7: Metrics for data augmentation applied on the final dataset—the threshold at 50%. ..	45
Table 8: Metrics for preliminary models applied on the final dataset. The threshold at 50%. ..	46
Table 9: Metrics achieved for wrapped and unwrapped interferograms patches. The threshold at 50%.....	48
Table 10: Metrics for tests on full interferograms	51
Table 11: Segmentation Metrics	52
Table 12: Metrics of segmentation for the best thresholds (90% for InceptionV3 and 70% VGG19)	54

Acronyms

AI – Artificial Intelligence

ANN – Artificial Neural Networks

CNN – Convolutional Neural Network

COMET – Centre for Observation and Modelling of Earthquakes, Volcanoes, and Tectonics

DEM – Digital Elevation Model

DL – Deep Learning

ESA - European Space Agency

GAM – Global Atmospheric Models

InSAR – Interferometric Synthetic Aperture Radar

IoU – Intersection Over Union

LiCS – Looking Inside the Continent from Space

ML – Machine Learning

NN – Neural Networks

RADAR – RAdio Detection And Ranging

RAR – Real Aperture RADAR

ReLu - Rectified Linear Units

ResNet – Residual Network

SAR – Synthetic Aperture RADAR

SLC – Single Look Complex

1. Introduction

1.1. Context and Motivation

Earthquakes are caused by a release of energy from faults like the tectonic plates, where blocks of crustal rocks move against each other, provoking a movement on the surface. The study and understanding of these natural disasters are critical, as they are responsible for the death of thousands of people and billions of euros of loss in the last decade. A single event can be catastrophic. For example, the 9.1 magnitude Haiti earthquake occurred in 2010, provoked more than 200,000 deaths and more than a million homeless people, achieving a loss of between six and thirteen million euros (Hou & Shi, 2011). A more recent case was the 6.1 Italy earthquake, occurred in 2016, killing 226 people (Sorrentino et al., 2019). Scientists can not predict earthquakes, but techniques allowing a response as fast as possible can be developed, providing some seconds or even minutes before the event achieves civilisation (Dong & Shan, 2013).

Due to the satellites' capability to obtain a considerable amount of information for big areas in a short time, they have been used to study the globe. In particular, Space-borne Synthetic Aperture RADAR (SAR) sensors can obtain surface information 24h per day regardless of weather conditions. SAR interferometry (InSAR) represent a set of methods that can be used for studing. Two SAR images captured at different times for the same area permit the creation of interferograms that register deformations in the order of centimetres (or less) – it is in the scale of half of the wavelength used. Thus, the first practical demonstration of Space-borne InSAR consisted on the use of ERS-1 interferogram to map the deformations of the 1992' Landers earthquake (Gombert et al., 2018). Since that time, the developments in InSAR techniques and the surprising results they allowed to achieve, transformed these methods into a mature technology in the detection of deformations. The success achieved by InSAR led space agencies, both governamental and private, to develop and launch new SAR missions, resulting in a more extensive amount of data from different bands and resolutions. The success of ESA's ERS-1/2 followed by Envisat led ESA to launch C-band Sentinel-1 mission. It's medium resolution sensor characterises this mission with a 250km-wide swath, global coverage, and a short revisit time (6 or 12 days). Sentinel-1 Level-1 products are available free of charge, which represents a milestone in InSAR technology. Scientists for all around the globe can now access the huge amount of SAR data to study natural and man-influenced deformations and improve this technology. Therefore, Sentinel-1

data has been used for the study of landslides, glaciers, aquifers, and other natural and man-caused hazards, but having a particular success in the study of volcanoes and earthquakes, leading not only to the evolution of information of these catastrophes but also for the evolution of InSAR. In fact, InSAR technology become a global monitoring tool in the last decade. All this evolution brings a new problem: how to deal with such amount of data? Nowadays, InSAR data fits in the domain of big data. For example, a single Sentinel-1 satellite generates more than 10 Tb of data per day. Therefore, with all this data it is no longer possible to process and analyse this data using conventional and hand-based methods. Thus, it is mandatory to find a way that automatically detects deformation to fully explore ESA (and other space agencies) archives (Potin et al., 2019).

The scientific community has been developing new methods to deal with big data problems, and machine learning stood out as one of the most used. The evolution of technology and the appearance of new methods, allowed a revolution of artificial intelligence techniques, being Machine Learning and Deep Learning among the best successful. These techniques have been fulfilling what they promised. Machine Learning automates human processes for a large volume of data dealing with speech recognition, language translation, and image classification problems. It is nowadays one of the most used techniques by data scientists. Artificial neural networks are a Machine Learning technique and consist of trained models that pass the data through hidden layers to predict the response for a given problem. With the evolution of the neural networks, these models have more layers, achieving even better results. The layers evolved, and new types were created, like the convolutional layers that are 2D tensors. The introduction of convolutional layers into the neural network originates the modern convolutional neural networks (CNN). They become very successful for image classification problems. Scientists and engineers already make CNN architectures with high accuracy in plenty of datasets applied to specific cases.

1.2. Objectives

This work's main objective aims to explore Machine Learning techniques for automatic detection of seismic deformations in SAR interferograms and estimate the earthquake's impact area. To achieve this objective, we started by creating and annotating a dataset for training Deep Learning models. Then, we explored Deep Learning classifiers to detect seismic fringes in interferogram patches. Finally, we explored Deep Learning segmentation models to identify the seismic-fringes pixels in the interferogram. This allowed us to estimate the areas affected by the earthquakes.

During this work, we looked to answer the following questions:

- Are the seismic deformations equally detected in wrapped and unwrapped interferograms?
- What is the best state-of-the-art Deep Learning models to detect seismic deformations in SAR interferograms and estimate the earthquake's impact area?
- Although vast public data are available on the COMET website, dataset creation requires a long time and effort to annotate. Thus, with the small dataset possible to create during this work, how efficiently is it possible to train Deep Learning models?
- Available data is very unbalanced – there many areas without fringes and few with fringes. How to deal with this unbalanced and improve models training?

1.3. Publications

Three posters were published during the realisation of this project, and a conference paper (Scopus-indexed) is already accepted:

- E-poster video Fringe 2021: “Detect Earthquake In SAR Interferograms Using Machine Learning.”; **Bruno Silva**, Joaquim Sousa, António Cunha.
- E-poster for Dragon Symposium 2021: “Unbalanced Technics to Improve the Train for ML Models to Detect Earthquake Fringes”; **Bruno Silva**, Joaquim Sousa, Milan Lazecky, António Cunha. (Attachment 1)
- E-poster for Dragon Symposium 2021: “ML Segmentation Models to Automatically Identify Areas Affected by Earthquakes”; **Bruno Silva**, Joaquim Sousa, Milan Lazecky, António Cunha. (Attachment 2)
- Conference paper for Centeris 2021: “Deformation Fringes Detection in SAR interferograms Using Deep Learning”; **Bruno Silva**, Joaquim Sousa, Milan Lazecky, António Cunha. Proceedings will be published in Scopus-indexed (Elsevier Procedia Computer Science Journal).

1.4. Structure of the Document

Five chapters are present in this document. After this introductory chapter where the motivation and main objectives are presented, in Chapter 2 a **literature review**, focus on the concepts that support the work is presented. The chapter starts with InSAR and deep learning fundamentals, and ends with the explanation of some CNN architectures and some techniques used in this work. **Material and methods**, which describe how the work is done step-by-step are discussed in Chapter 3 and the main **Results achieved and their discussion** is presenting the metrics achieved by the trained models and the interpretation, are presented in Chapter 4. Finally, Chapter 5, presents the main **Conclusions by answering the research question stated in Chapter 1**.

2. Literature review

For the realisation of this work, it was necessary to understand two subjects. The InSAR data used in this work, how it works and what we can get from it, and Machine Learning, the technique we use to achieve our goal.

Here is presented what we found in the literature, which is relevant to understand this work. To start, we will introduce the InSAR fundamentals, followed by the machine learning fundamentals. Then we go deeper into machine learning and theoretically explain the methods and architectures we use to detect the earthquakes fringes. Finally, we present and analyse the studies where the authors use deep learning to detect fringes in SAR interferograms.

2.1. InSAR fundamentals

This section is the explanation of InSAR, the data used in this project. It starts with an introduction to the SAR sensor, the sensor responsible for interferograms. Then, it explains the interferograms, how they are obtained, the possible information, and how to distinguish earthquakes of the remaining deformations. Finally, is the introduction of Sentinel-1, the satellite responsible for the data in this project.

2.1.1. Synthetic Aperture Radar

Synthetic-aperture radar (SAR) is a form of radar used to create two-dimensional images or three-dimensional reconstructions of objects, such as landscapes (Kirscht & Rinke, 1998) and reconstruction urban environment (Balz & Haala, 2002). For that, SAR systems transmit electromagnetic waves at a wavelength that can range from a few millimetres to tens of centimetres (Table 1).

Table 1: The various frequency bands.

SAR Band	Frequency Range (GHz)	Correspondent wavelength range(cm)
P	0.230-1	130-30
L	1-2	30-15
S	2-4	15-7.5
C	4-8	7.5-3.75
X	8-12.5	3.75-2.40
Ku	12.5-18	2.40-1.67
K	18-26.5	1.67-1.13
Ka	26.5-40	1.13-0.17

These waves do not reflect on the atmosphere. Furthermore, they do not need sunlight to work appropriately (active sensor), so they can operate during the day and night under all weather conditions (Curlander & McDonough, 1991).

Despite the wide variety of bands, the most used bands on-board SAR satellites are the L, C, and X bands. These sensors are usually attached to a moving satellite. Consequently, the sensor emits a sequence of electromagnetic pulses that send the echo back to the sensor. The echo is received and recorded when reflecting on the earth. A "synthetic aperture is created" when all the received signals are combined, providing a much-improved azimuth resolution than the Real Aperture Radar (RAR) (Figure 1).



Figure 1: Illustration of a space-borne SAR sensor. Source:(Raab, 2016)

After the recording process, the sensor obtains two pieces of information. The time it takes the signal to reflect on the surface and return to the satellite and the signal intensity. Knowing the travel time of the signal(t) and its speed(c) (speed of light in the vacuum), it is possible to calculate the distance from the satellite to the ground in the following way:

$$Distance(D) = \frac{c * t}{2} \quad (1)$$

SAR images are created with the signal intensity (or signal amplitude). For that, the intensity of the signal can be modeled by the following equation (Nicolas & Adragna, 2000):

$$Pr = Pe \frac{G^2 \lambda^2 \sigma^2}{(4\pi)^3 D^4 a} \quad (2)$$

Where:

Pr: received power.

Pe: Transmitted power.

G: antenna gain.

λ : wavelength.

a: losses related to absorption in the propagation medium.

D: range between antenna and target.

In the last years, SAR has been used for several studies in the most varied areas such as agriculture (LIU et al., 2019), forestry (Padalia & Yadav, 2017), hydrology (Lang & Kasischke, 2008), oceanography (Marino et al., 2016) and security (Tello et al., 2005).

2.1.2. Interferometric Synthetic Aperture Radar and earthquakes

SAR sensors can also store the signal phase and the amplitude, a value between $-\pi$ and π that represents the signal at the moment of receipt by the sensor. Obtaining at different times two SAR images, it is possible to create an interferogram $\varphi(x,r)$ calculating the phase difference between them (Chen et al., 2000):

$$\varphi = \Delta\Phi = \Phi_1 - \Phi_2 = \frac{2\pi Q \delta R}{\lambda} \quad (3)$$

Where:

Φ = phase.

Q: $Q = 1$ when the two antennas are mounted on the same flying platform; otherwise, $Q = 2$.

δR = slant range difference.

λ : wavelength.

Through the interferogram, it is possible to create a digital elevation model (DEM). However, when topography is eliminated, deformations that occurred during the period of the two phases with precisions in the order of half the wavelength are observed (Figure 2).

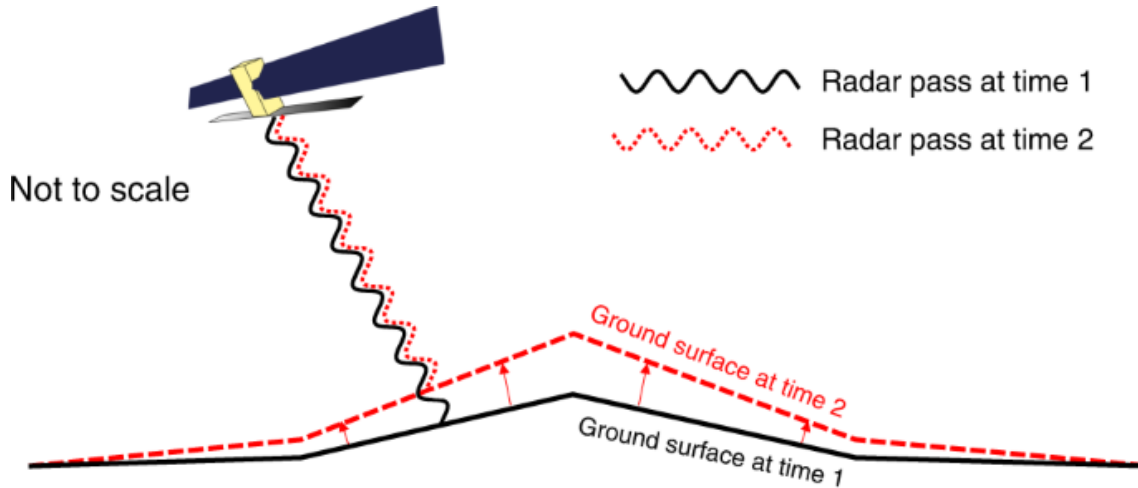


Figure 2: INSAR functioning, source: (Biggs & Wright, 2020)

The phase values are radian (between $-\pi$ and π), not making its application very practical. To overcome the radians problem, it is necessary to unwrap the phase changing to a metric scale, facilitating its applicability (Figure 3).

a) Wrapped interferograms

b) Unwrapped interferogram

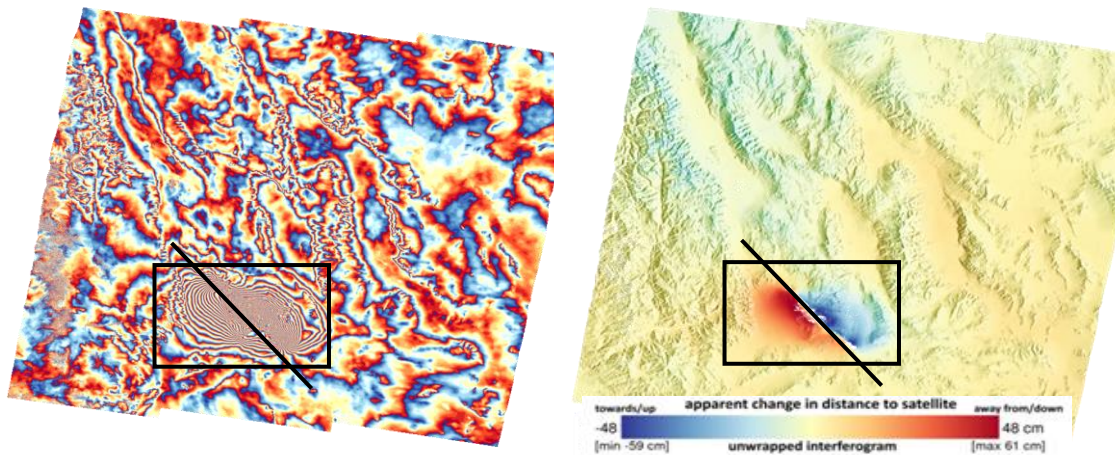


Figure 3: Interferograms from an earthquake in California (2019). Inside the box, it is possible to see the deformation area as the fault represented by the line: (Figures obtained in LICSBAS database)

In wrapped interferograms, deformations are characterised by fringes (Figure 3a), where each fringe corresponds to a deformation in the order of half of the signal's wavelength. These fringes can have different sizes and intensities, depending on the deformation scale, the difference in the dates of the two SAR images, or the amount of

atmospheric noise. Unwrapped interferograms are usually associated with a color scale that represents the displacement value. The greater the deformation, the greater the loss of information in the unwrapped interferograms.

What makes earthquakes different from other types of deformation in interferograms is that it is usually possible to see the fault that causes the earthquake. However, this fault may not be visible when dealing with more minor deformations.

InSAR has been used to monitor deformation in the earth, with a particular interest in volcanology (Ofeigsson et al., 2011) and earthquakes (Moro et al., 2017), but also inland subsidence (Chaussard et al., 2014), landslide (Schlögel et al., 2015) and Glaciaries motion monitoring (Pavelka et al., 2019).

2.1.3. Sentinel-1

As part of the European Copernicus program from the European Space Agency (ESA), the Sentinel-1 mission, based on a constellation of two SAR satellites, ensures continuity for ERS/Envisat C-Band SAR observations for Europe. The mission is characterised by large-scale, frequent, and repetitive observations, systematic production, and a free and open data distribution policy (Attema, 2005). The Sentinel-1 mission includes four exclusive imaging modes with different resolutions (down to 5 m) and coverage (up to 400 km). It provides a dual-polarisation capability (HH+HV, VV+VH)(H= horizontal; V= vertical), short revisits times, and rapid product delivery. For each observation, precise spacecraft position and attitude measurements are available (Potin et al., 2019). This satellite operates in four exclusive acquisition modes: Stripmap (SM), Interferometric Wide swath (IW), Extra-Wide swath (EW), Wave mode (WV) (Figure 4).

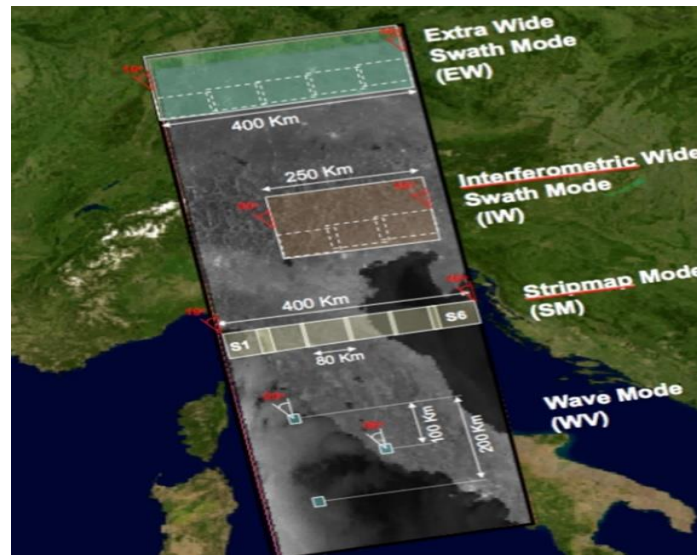


Figure 4: "The SM, IW, and EW modes support operation in single selectable polarisation (HH or VV) and dual-polarization (HH+HV, VV+VH), implemented through one transmit chain (switchable to H or V) and two parallel receive chains for H and V polarisation. The WV mode only supports single selectable polarisation (HH or VV)". (source: <https://sentinel.esa.int/web/sentinel/technical-guides/sentinel-1-sar/sar-instrument/acquisition-modes>, 26/03/2021)

Sentinel-1 provides three product levels. Level-0 products are the basis from which all other high-level products are produced. Level-1 focused data are the products intended for most data users. The Level-0 product (raw data) is transformed into a Level-1 product by the Instrument Processing Facility (IPF) via the application of various algorithms, and data can be processed into either Single Look Complex (SLC) and Ground Range Detected (GRD) products. There is also a Level-2 that consists of geolocated geophysical products derived from Level-1. The data used in this work are SLC products from Level-1. These are images in the slant range by azimuth imaging plane, in the image plane of satellite data acquisition. Each image pixel is represented by a complex magnitude value and therefore contains both amplitude and phase information. SLC images are produced in a zero Doppler geometry (N.Miranda et al., 2015; Vincent et al., 2020).

The ground segment provides systematic generation and availability of Level-1 SLC products for all data acquired over land and ice masses since July 2015. Such SLC production increase has fostered the exploitation of Sentinel-1 data for an increasing number of InSAR applications and over a growing number of areas worldwide (Potin et al., 2019).

2.2. Machine learning

This section introduces the fundamentals of deep learning (DL). DL is a type of machine learning method and belongs to the artificial intelligent area. It is an evolution of neural networks (Figure 5).

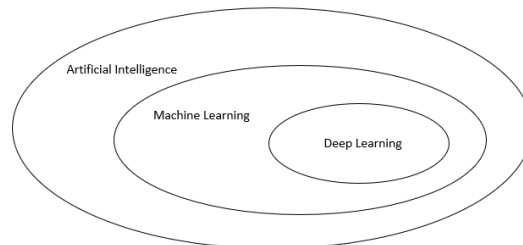


Figure 5: Evolution of artificial intelligence

2.2.1. Artificial Intelligence

Artificial Intelligence (AI), when it appeared in the 1950s, aimed to automate human processes, like playing chess. There was no "learning" in the beginning, only an involved hardcoded rules crafted by programmers. This type of "intelligence" cannot solve complex problems such as speech recognition, language translation, and image classification. Knowing this problem, experts wondered if it would be possible for the computer to tell beyond what man knows how to perform, this way born Machine Learning (François Chollet, 2018).

Unlike classic programming that uses data and rules to get an answer, machine learning uses data and answers to create the rules (Figure 6).

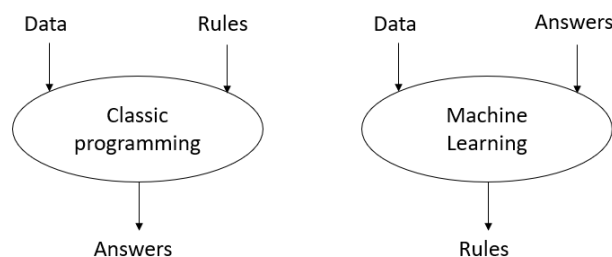


Figure 6: Machine Learning creates a new way of looking at data. This type of system is trained rather than explicitly programmed (adapted figure from Chollet, 2018)

There are various approaches of ML, like:

- **Supervised classifier**, where the algorithm uses data and labels to create a function that maps inputs to desired outputs.
- **Unsupervised classifier**, where labeled examples are not available.
- **Semi-supervised classifier**, a combination of the previous two.

- **Reinforcement learning**, where the algorithm learns how to act given an observation of the world.
- **Transduction**, similar to supervised learning, but does not explicitly construct a function: instead, tries to predict new outputs based on training inputs, training outputs, and new inputs,
- **Learning to learn**, the algorithm learns its own inductive bias based on previous experience (Ayodele, 2010).

In this case, the focus is supervised learning, the one used in this work.

2.2.2. Artificial Neural Networks

Artificial Neural Networks are based on neurons in the mammalian cortex and are modeled loosely but on a much smaller scale. Neural Networks are a network of multiple layers of neurons consisting of nodes used to classify and predict data provided some data as input to the network. Thus, there is an input layer, one or many hidden layers, and an output layer. All the layers have nodes (also called neurons), and each node has a weight and bias that is considered while processing information from one layer to the next layer (Sharma et al., 2020) (Figure 7).

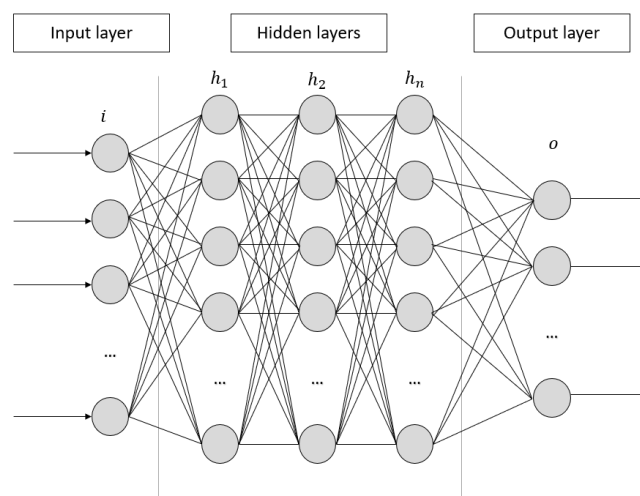


Figure 7: Representation of a Neural Network (adapted figure from Sharma et al., 2020)

During the training of an NN, each node's weight and bias are constantly updated in each epoch, improving the output and understanding the data. Furthermore, an activation layer is defined to enhance the space of hypotheses in each layer. For classification problems and consequently for segmentation, ReLu is usually used as activation layer in the intermediate layers, and for the output layer, sigmoid or softmax is used.

Relu (Agarap, 2019) eliminates negative values in input data, turning all negative values into zero and keeping positive values (Figure 8a). The sigmoid function (Narayan,

1997) uses function (4) that gives a value between zero and one (usual, a probability) (Figure 8b).

$$Sigmoid(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

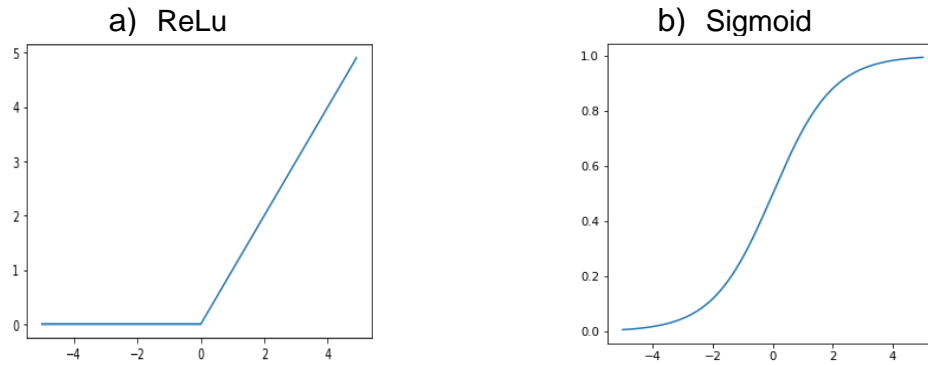


Figure 8: Activation functions a) ReLu; b) Sigmoid

The softmax function has a behavior identical to the sigmoid, but instead of a value, it gives N values between 0 and 1, where N = numbers of existing classes.

The diagram of Figure 9 shows how is the training of NN layers.

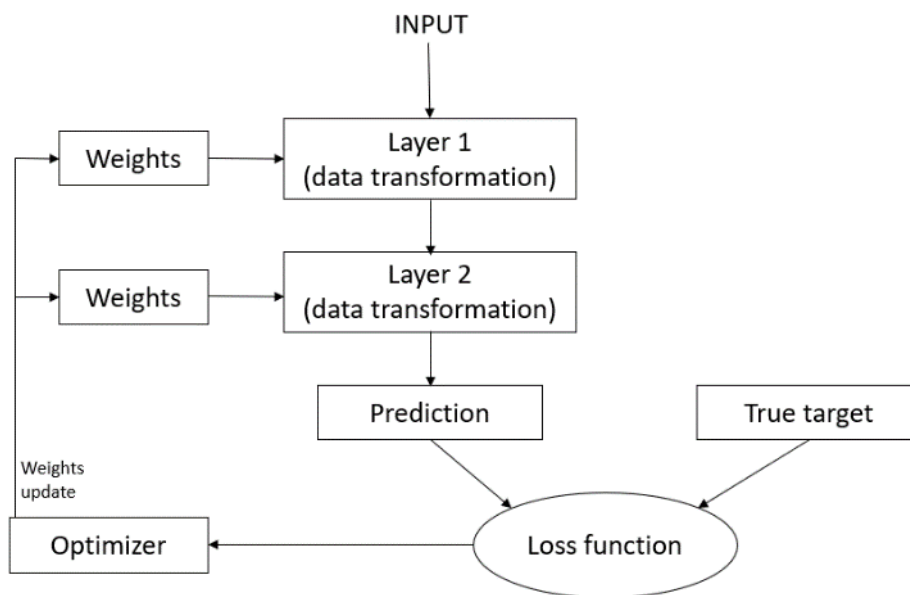


Figure 9: Usual architecture of a neural network (adapted figure from Chollet, 2018).

To start, it is needed to prepare the data that will be the input. Then, the data will pass for the layers, giving a response (for the first time, it will be a random one). After that, it will compare the response provided for the algorithm and the actual response given by the operator (on the input), the difference will be the loss score. Finally, the loss

score will be used as a feedback signal to adjust the value of the layers' weights a little, in a direction that will lower the loss score for the current example. This adjustment is the job of an optimiser. This process will happen successively until the model obtains the best possible answer (François Chollet, 2018). The regularisation techniques appear in the form of layers and can improve the evolution of the model through the best answer possible.

The operator's job is to choose the number and type of the layers and what is "inside the boxes" of Figure 9.

Loss Function

When training the model, the initial output will be a random value. Then this value will be "subtracted" from the desired real value through the loss function so that throughout the training, the output will be closer to the expected value (Chollet, 2018). Choosing the proper loss function for the right problem is extremely important: the network will take any shortcut to minimise the loss. If the objective does not fully correlate with the task's success, the network will do unwanted things. In this case, as the output is a sigmoid, the probability of the input having deformation fringes, the indicated loss function will be binary crossentropy (Janocha & Czarnecki, 2016).

Binary crossentropy is a loss function that is used in binary classification tasks and is represented by the following formula:

$$Loss = -\frac{1}{N} \sum_{i=1}^N y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i)) \quad (5)$$

Where:

y_i = label (0 or 1)

$p(y_i)$ = predicted probability given by the model.

N = output size

Reading this formula tells that, for each input $y=1$, it adds $\log(p(y))$ to the loss, that is, the log probability of it being 1. However, conversely, it adds $\log(1-p(y))$, that is, the log probability of 0, for each input $y=0$.

Optimisers

Selecting an optimiser is a significant step in the current deep learning pipeline. The optimisation algorithm chosen by a deep learning practitioner determines the training speed and the final predictive performance of their model. After having the value of the loss score, the optimiser is responsible for adapting the value of the weights. To date, no theory adequately explains how to make this choice. Instead, the choice relies on

empirical studies (Wilson et al., 2017) and benchmarking (Schneider et al., 2019). Analysing some studies, Choi et al., 2019 show that more general optimisers (like RMSProp, or Adam) never underperform their special cases (like Stochastic gradient descent (SGD) or Nesterov). Looking at the results, although the chosen hyperparameters can alter the results, Adam has slightly better results in most cases. For this reason, Adam is used in this work.

Adam (Kingma & Ba, 2015) is an adaptive learning rate method that computes individual learning rates for different parameters. It uses the advantage of momentum (6) by using the moving average of the gradient like SGD (7 with momentum and uses the squared gradients to scale the learning rate (9) like RMSProp (8).

$$w_{t+1} = w_t - \alpha m_t \quad (6)$$

Where:

$$m_t = \beta m_{t+1} + (1 - \beta) \left[\frac{\delta L}{\delta w_t} \right] \quad (7)$$

Where:

w_t = weights at time t

m_t = aggregate of gradients at time t (initially, $m_t = 0$) (momentum)

α_t = learning rate at time t

δL = derivative of Loss Function

β = Moving average parameter (const, 0.9)

$$w_{t+1} = w_t - \frac{\alpha_t}{(v_t + \epsilon)^{\frac{1}{2}}} * \left[\frac{\delta L}{\delta w_t} \right] \quad (8)$$

Where:

$$v_t = \beta v_{t+1} + (1 - \beta) \left[\frac{\delta L}{\delta w_t} \right]^2 \quad (9)$$

Where:

v_t = sum of the square of past gradients (initially, $v_t = 0$).

ϵ = A small positive constant (10^{-8})

m_t and v_t have both initialised as 0 (based on the above methods), it is observed that they gain a tendency to be 'biased towards 0' as both β_1 & $\beta_2 \approx 1$. This optimiser fixes this problem by computing bias-corrected m_t and v_t . This is also done to control the weights while reaching the global minimum to prevent high oscillations when near it. The formulas are:

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t}, \widehat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (10)$$

Now, instead of normal weight parameters m_t and v_t , it takes the bias-corrected weight parameters. Then, putting them into the ADAM's we get the general formula.

$$w_{t+1} = w_t - m_t \left(\frac{\alpha_t}{\sqrt{\hat{v}_t + \epsilon}} \right) \quad (11)$$

Regularisation techniques

During the training of a neural network, after some epochs, the training becomes inefficient. This is called overfitting and is characterised by an excessive increase in loss. However, some techniques to overcome this problem include L1 and L2 regularisation, Dropout, and batch normalisation.

L1 and L2 regularisation (Ng, 2004), consisting of adding weight to the loss, make the distribution of values more regular. In L1 regularisation, the cost added is proportional to the absolute value of the weight coefficients. In L2 regularisation, the cost added is proportional to the square of the value of the weight coefficients.

Dropout (Srivastava et al., 2014) takes a percentage (defined by the operator) of the output values of a particular layer and transforms them into 0. This technique consists of creating a noise that allows reducing the fit of the training data.

Batch normalisation (Ioffe & Szegedy, 2015) is an algorithmic method that makes Neural Networks training faster and more stable. It consists of normalising activation vectors from hidden layers using the current batch's first and second statistical moments (mean and variance).

2.2.3. Deep learning

The "deep" in deep learning stands for the "idea of successive layers of representations. How many layers contribute to a model of the data is called the depth of the model" (Lecun et al., 2015). The NN was evolved in-depth and with the addition of new layers, giving rise to the Convolutional Neural Network (CNN). CNN is a class of ANN used successfully to process and analyse digital images. Like other Neural Networks, Convolutional Neural Networks has an input, an output, and hidden layers. The difference is that the hidden layers are essential 2D Convolutional layers (Conv2D), 2D tensors (Y. Lecun et al., 1998).

CNNs are capable of find patterns in images. These layers use filters in convolutional layers, leading to a sharp increase in the amount of data processed in the network. In addition, the pooling layer of max-pooling in some considered neighborhoods is used to reduce them (Valueva et al., 2020) (Figure 10).

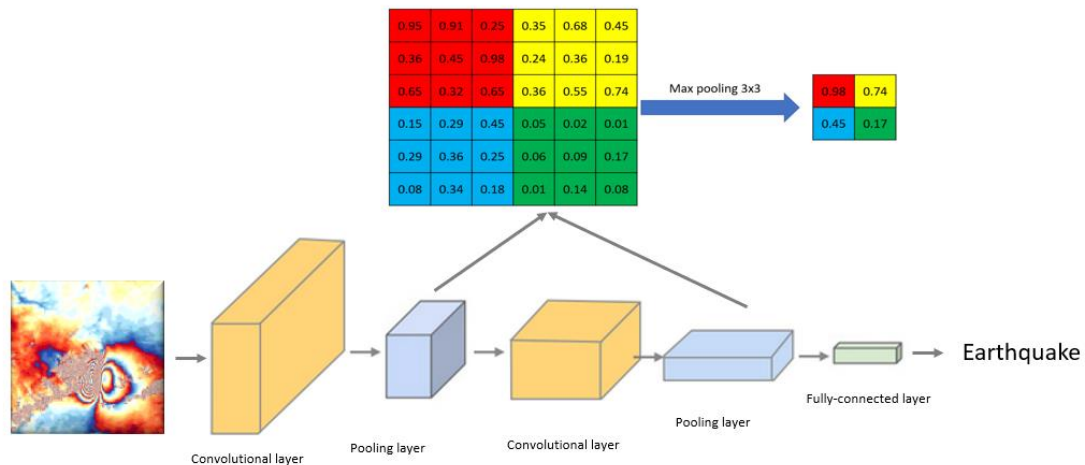


Figure 10: Example of the architecture of CNN with max-pooling operation using a mask of 3x3 matrix.

In terms of performance, convolutional neural networks outperform neural networks in terms of conventional image recognition tasks. Also, CNN's are more efficient in memory and complexity and have better features extraction capability, better for entirely new tasks. This means that CNNs can extract useful attributes from an already trained CNN with its trained weights by feeding your data on each level and tune the CNN a bit for the specific task.

2.3. Deep learning methods

One of the biggest challenges encountered during this work was the lack of data and the unbalanced data. Interferograms with deformation fringes were more abundant than those without them. A widely used technique to deal with that problem is the use of pre-trained networks. Bellow, we present some architectures used for classification and segmentation tasks.

2.3.1. Architectures for classification

There is a wide variety of pre-trained CNN architectures for image classification that can be used. Here is explained those used in this project.

InceptionV3 is a model that belongs to the inception family. The objective of Inception is to improve the detection of objects that can occupy different size areas in the image, winning the 2015 ImageNet challenge. Therefore, they created a more "wider" model instead of deeper, using filters of different sizes (1x1, 3x3, and 5x5) working at the same level that are connected in the end (Figure 11)(Szegedy et al., 2015). Later for

the Inception V2, the 5x5 convolutional was transformed into two 3x3 convolution operations to improve computational speed. Finally, for the third version of Inception, the 7x7 convolutionals were factorised, batch normalisation was used for the auxiliary classifiers, and it was applied label smooth, improving the model without drastically changing the modules (Szegedy et al., 2016).

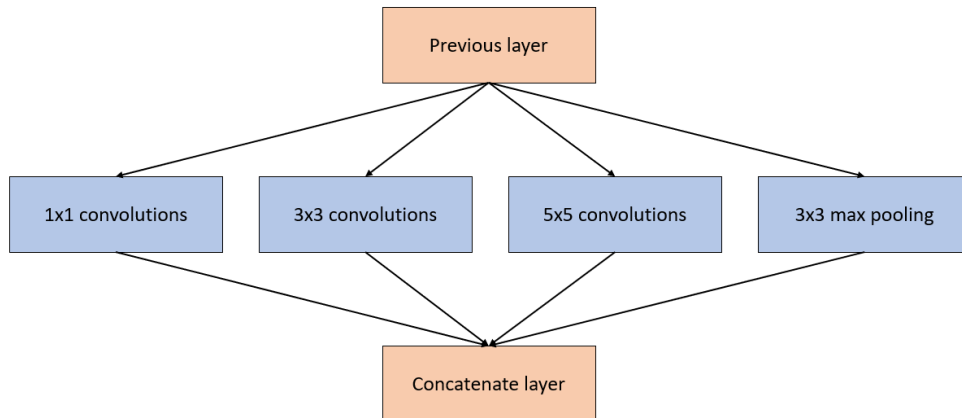


Figure 11: Idea behind inception model. Adapted figure from Szegedy et al., 2015

2 VGG models won first and second place in Imagenet challenge 2014. They are VGG16 and **VGG19**. The numbers (sixteen and nineteen) represent the number of weight layers in the network. These are very deep convolutional networks. VGG19 comprises 16 convolution layers 3x3, three fully connected layers, five MaxPool layers, and one SoftMax layer (Figure 12)(Simonyan & Zisserman, 2015).

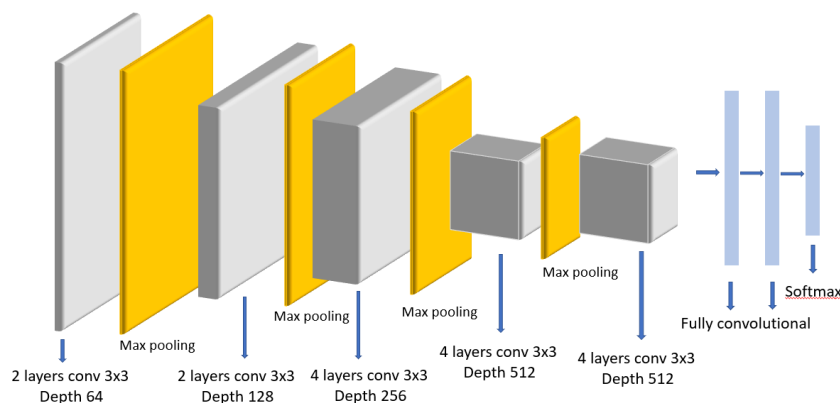


Figure 12: Architecture of VGG19

ResNet made it possible to train hundreds of layers and still achieve a good performance. Increase the deepness of the model is good up to a point. After that, if added more layers, the accuracy drops. To overcome this problem, the authors of ResNet apply a so-called "identity shortcut connection" that skips one or more layers.

This way, stacking more layers, do not reproduce worse results. This architecture is also composed of Batchnormalization layers and does not have fully connected layers (He et al., 2016a). Later, the same authors created improvements to the ResNet architecture, calling it **ResNetV2** (He et al., 2016b) (Figure 13).

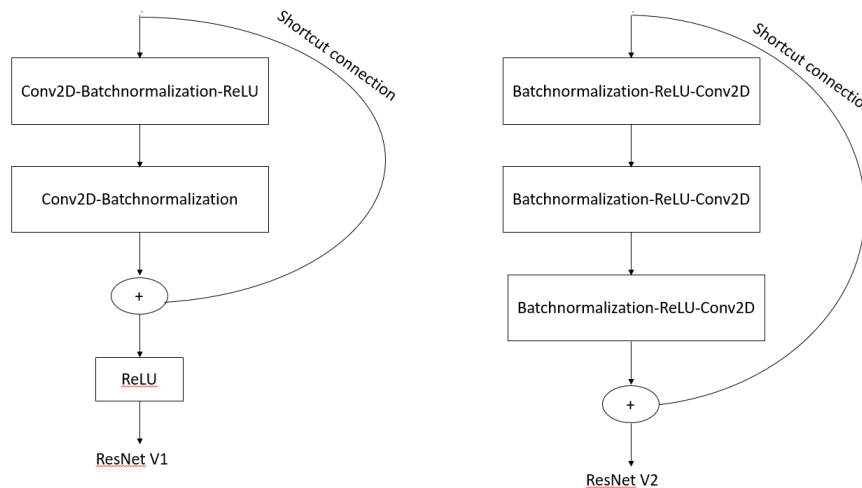


Figure 13: A residual block of ResNet and ResNetV2. Adapted image from (Van Hieu & Hien, 2020)

2.3.2. Architectures for segmentation

CNN's are also used for segmentation. Instead of classifying an image, the model classifies each pixel individually. Likewise, it is needed to give the real response for each pixel (a mask) in the input. Therefore, the output must be a matrix of the size of the desired image. There are some segmentation techniques like U-net and dual attention

Unet (Weng & Zhu, 2015) comprises two parts, an encoder and a decoder. The encoder is a regular CNN and can use one of the state of the art or be pre-trained by us. This part can also be frozen or unfrozen, depending on if it wanted to train the encoder or not. The encoder is responsible for capturing the context of the image. The decoder will do the opposite. It will be the encoder in reverse, where the output will be the desired image size, usually the same size as the input. Drawing the architecture, this model looks like a "U" (Figure 14).

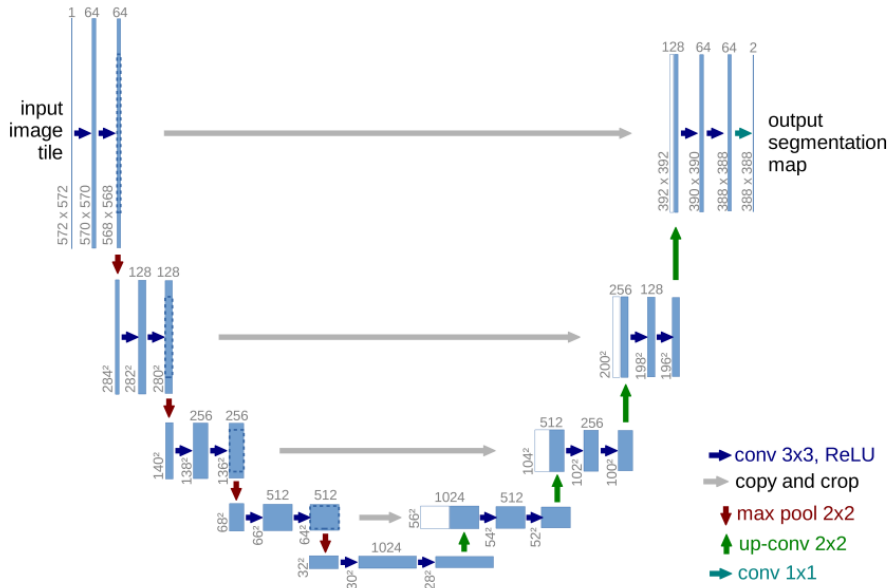


Figure 14: Architecture of Unet. Credit: Weng & Zhu, 2015

Dual attention is a segmentation architecture that uses the resnet architecture followed by two segmentation modules (position attention module and channel attention module) to achieve better segmentation. The position attention model is responsible for detecting the location of objects, generates a spatial attention matrix that models the spatial relationship between any two pixels of the element. The channel attention module detects color channel differences and creates a channel attention matrix for model interdependencies between channels. The authors create this architecture for scene segmentation (Fu et al., 2018)(Figure 15).

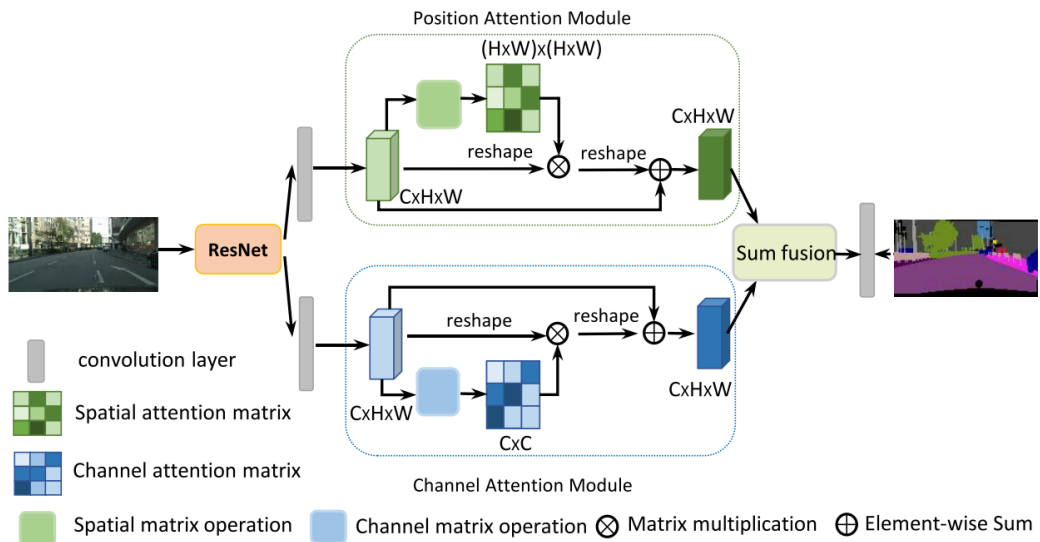


Figure 15: Dual attention architecture. credit: Fu et al., 2018

2.4. Deep Learning for InSAR

"InSAR techniques have been known and used for decades, but it has gone from being an opportunistic science to becoming a global monitoring tool in the last decade" (Biggs & Wright, 2020).

The use of machine learning in InSAR is recent, so studies in this area are scarce. For example, we only found one study where scientists use artificial intelligence to classify earthquake fringes from SAR interferograms. But for volcanic deformations and slow ground deformations in SAR interferograms, there are some more studies.

2.4.1. Earthquake deformations detection

In this study, (Brenngman & Barnhart, 2021) created a model called SarNet to detect earthquake fringes and used 4 million synthetic interferograms to train the model. Although the model obtained an accuracy of 99.74% in the validation data (synthetic data), this value drops to 85.22% when tested on a dataset with real data. The authors also used Class activation maps to guarantee that the model returns the location of the fringes.

2.4.2. Volcanoes deformations detection

The first study applied machine learning to classify volcanic deformation (N. Anantrasirichai et al., 2018). In this study, the dataset is composed of 30,249 Envisat and Sentinel-1 wrapped interferograms from approximately 900 volcanoes worldwide. The data was unbalanced with 300 positives samples and over more than 100 times more negatives samples. Therefore, the authors created positive samples through data augmentation to compensate for the imbalance, applying a slight distortion to the images to balance the data. Next, they train the data in three pre-trained CNNs: AlexNet (Krizhevsky et al., 2012), ResNet50 (He et al., 2016b), and InceptionV3 (Szegedy et al., 2016), and an SVM classifier based on textural features (N. Anantrasirichai et al., 2013). AlexNet proved to have better accuracy, with 0.995 against 0.989 from ResNet50, 0.975 from InceptionV3, and 0.968 from SVM. In the end, another test was made but only in AlexNet and SVM. In this case, interferograms were tested from two volcanoes separately (Erta Ale and Etna). Once again, CNN achieved better results with an accuracy of 0.994 and 0.871 against 0.985 and 0.742.

With the need to have balanced data, the same authors applied deep learning to detect deformation in volcanoes but used a new way of synthetic data (N. Anantrasirichai

et al., 2019). "Has been created Synthetic data of the deformation patterns based on a Monte Carlo selection of parameters for forwarding analytic models, stratified atmospheric effects derived from weather models, and turbulent atmospheric effects based on statistical simulations of correlated noise." The results show that adding synthetic data gives better results than using only real data, reducing the number of interferograms that required manual inspection by half and decreasing false positives by >80%. Another study where synthetic data is used to create a network can detect deformations in interferograms (Valade et al., 2019). In this case, the objective is to monitor volcanoes using multisensor sentinel missions and artificial intelligence.

The last study found to date was also automatic detection of volcanic surface deformation but this time applied to unwrapped interferograms (Sun et al., 2020). The network was trained with synthetic data and then tested with real data from Masaya volcano, Nicaragua. The authors demonstrated using an end-to-end CNN with an encoder-decoder architecture as a potential tool for globally practical near real-time volcanic unrest detection. The CNN can reveal noise-free surface deformation signals from unwrapped surface displacement maps with variant SNRs. However, the precision of detected results depends on the time scale that time-consecutive unwrapped surface displacement maps covered."

2.4.3. Slow ground deformation detection

In this area, Anantrasirichai et al. (2018) used Convolutional Neural Networks to Detect Slow, Sustained Deformation in InSAR Time Series. The authors use synthetic wrapped data and found that when the number of fringes is increased to double, the detection threshold is lowered by 25/30%. Thus, an estimated detection threshold was 3.9 cm for deformation signals alone and 6.3 when considered atmospheric artifacts. Overwrapping reduces to 1.8 and 5.2 cm, respectively. However, when data is overwrapping, the false positives increases. When applied to the study areas, a deformation of 8.5 cm/year was detected for Campi Flegrei and 3.5 cm/year for Dallol. This corresponds to cumulative displacements of 3 and 4 cm consistent with estimates based on synthetic data.

Two articles study the deformation of a place. One made for the Erhai region, China (Wang et al., 2019), and the other for the whole of the United Kingdom (Nantheera Anantrasirichai et al., 2020).

In the first one, Sentinel-1 Synthetic Aperture Radar (SAR) images "were used to study the characteristics of ground deformation in the Erhai region using the Small

Baseline Subset Interferometric SAR (SBAS-InSAR) technique." Then the backpropagation (BP) algorithm was used to predict and verify the ground deformation of the study area with high precision.

In the second one, the authors "explore the applicability of deep learning approaches by adapting a pre-trained convolutional neural network (CNN) to detect deformation in a national-scale velocity field." However, like in the other cases, they have insufficient ground truth data to construct a balanced training data set. As a result, the deformation signals are slower and more localised than in previous applications. To tackle this problem, the authors used three methods: "i) spatial interpolation with modified matrix completion, ii) a synthetic training dataset based on the characteristics of real UK velocity map, and iii) enhanced overwrapping techniques" (like already seen in previous studies this reduce the detection threshold). "The results demonstrate the potential applicability of the proposed framework to the development of automated ground motion analysis systems."

A study predicts InSAR time-series deformation using deep convolutional neural networks (Ma et al., 2019). First, the authors studied two types of deformation prediction, settlement prediction, and seasonal deformation prediction. For that, the authors detected the persistent scatterers (PS) and distributed scatterers (DS). With that, was used an interpolation method resulted in the training data for the DCNN. The results show that DCNN is very good at predicting deformation in a short time, with a 0.3 mm mean internal error.

To conclude, a study uses an auto-encoder architecture to detect deformation in the mm scale between the noise that remains after applying atmospheric correction from the GAMs. After applying the model to two real cases with success, one for slow deformation from a fault and the other for deformation caused by underground pressure changes, the model was capable of find deformation on a scale of 2mm in the first case and a scale of 5 to 7 mm in the second case (Rouet-Leduc et al., 2020).

2.5. Summary

In summary, we can obtain some essential points from the literature review to create our methodology.

For the data, we know that the medium precision C-band from the Sentinel-1 has a good precision for studying deformations created by natural disasters. We also know

that wrapped interferograms have more information when converted into a png image. However, they have more noise too.

For the method to detect the deformation fringes, artificial intelligence in general and deep learning, in particular, seems to be an excellent choice for the work. Not having a significant amount of data, we can use some pre-trained CNNs to help the train, obtaining a better result. There are several CNN's to use for the different objectives. Like VGG, that are known for being deeper to ResNet, there is more used to detect the different size of objects. We also demonstrate two architectures for the segmentation, one with an encoder-decoder (Unet) and the other using two segmentation techniques (Dual attention).

Analysing the literature, we realise that the studies are few and recent. However, we can already see the potential of deep learning applied to InSAR. For example, there is already a deep learning model to detect deformation created by earthquakes with some success. However, when we look for studies in other sources of deformation like volcanoes and slow ground deformation, we show that pre-trained state-of-the-art models obtain better results than the model created in the first study. Because of that, we will apply some of those pre-trained networks to detect fringes provoked by earthquakes.

3. Material and methods

This chapter presents the methodology devised to address the earthquake fringes classification and segmentation in InSAR tasks. This methodology took into consideration several issues pointed out in Chapter 2:

- The medium precision C-band from Sentinel-1 has information for studying deformations created by natural disasters.
- Wrapped interferograms have more information than unwrapped when converted into images, but they are noisier.
- Deep learning methodologies are adequate for detecting the deformation fringes (e.g., VGG and ResNet) and segmenting them (e.g., Unet).
- Small annotated datasets are challenging, but several approaches can help, such as transfer learning and data augmentation.

With all this in mind, a methodology to implement was proposed and is represented in the Figure 16 pipeline.

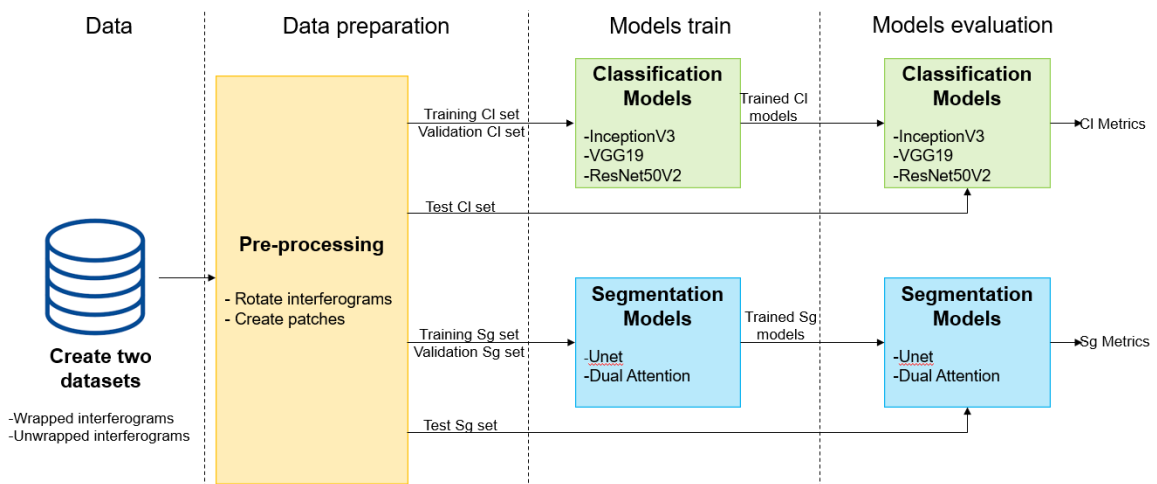


Figure 16: Proposed methodology pipeline.

The first step was creating and studying the InSAR datasets. Then, the data was normalized and manipulated to be prepared for the train, and the dataset was divided into three sets: train set, validation set, and test set. Next, the train and validation set to train both Classification and Segmentation models. Finally, the resulted models and the test set evaluate the models, achieving the metrics for the results. Each step of these steps will be described in detail in the following sections.

This work implementation used python version 3.7 with the following libraries:

- Tensorflow 2.3.0 and Keras 2.4.5 for machine learning;
- Segmentation_models 1.0.1 for the segmentation;
- Numpy 1.19.2 to prepare and manipulate the data;
- Sklearn (from Scikit-learn) 0.24.1 to calculate the metrics;
- Matplotlib 3.1.2 to create the graphs.

3.1. Data

To the best of our knowledge, there are no public datasets with annotated interferograms to be used with supervised machine learning methods. So it was decided to create one for this work by manually annotate wrapped and unwrapped interferograms without atmospheric correction from the public LiCS database from COMET Centre¹. This database made available satellite InSAR data from the Sentinel-1 mission since 2015.

First, a python program was created that automatically downloads all the available interferograms for a chosen area. This step was necessary because the website did not have a platform to download multiple data simultaneously and download the interferograms individually was not viable.

Twenty-nine zones were selected worldwide, where earthquakes occurred between June 2019 and May 2021. Figure 17 shows those zones in a planisphere.

¹<https://comet.nerc.ac.uk/comet-lics-portal/>

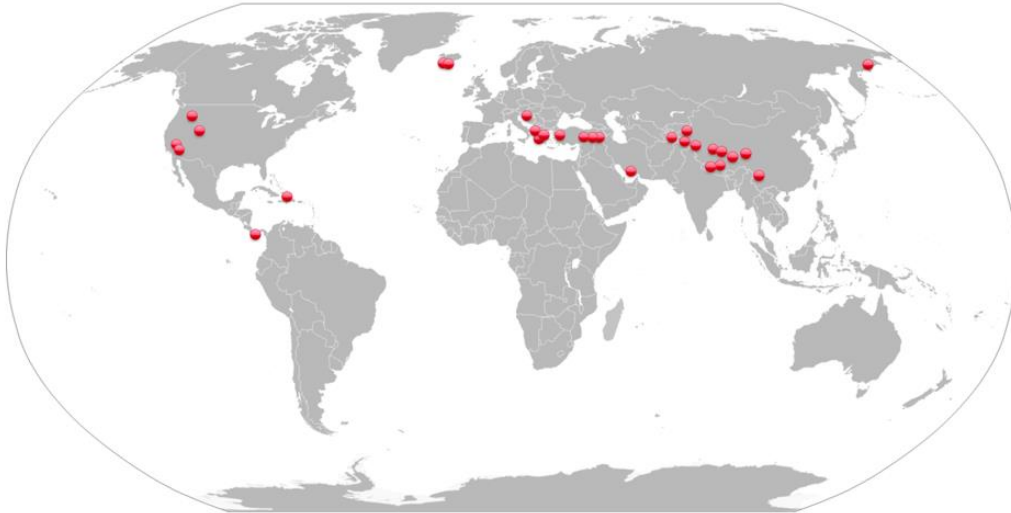


Figure 17: Earthquakes zones used in this work

Figure 17 shows that most of the earthquakes of this work occurred in the northern hemisphere. This choice was not deliberate. In fact, it happened because the southern earthquakes were not strong enough to cause deformation (e.g., Chile) or occur in zones with substantial noise where deformations are not visible in interferograms (e.g., Indonesia). Also is essential to mention that Sentinel-1 is a European mission, and consequently, data for Europe and the surrounding are more extensive.

A total of 3261 interferograms were downloaded, and 470 having positive deformation fringes are visible.

The dimension of the earthquake is mainly influenced by two factors, the magnitude in the Richter scale of the earthquake and the depth in which it occurs; the largest earthquakes usually occur at shallower depths in the earth's crust. However, it is essential to emphasise that the type of soil and the built environment can change the proportion of deformation.

In Table 2 (page 29), it can be seen the information about selected earthquakes and in Figure 18 their magnitude distribution histogram.

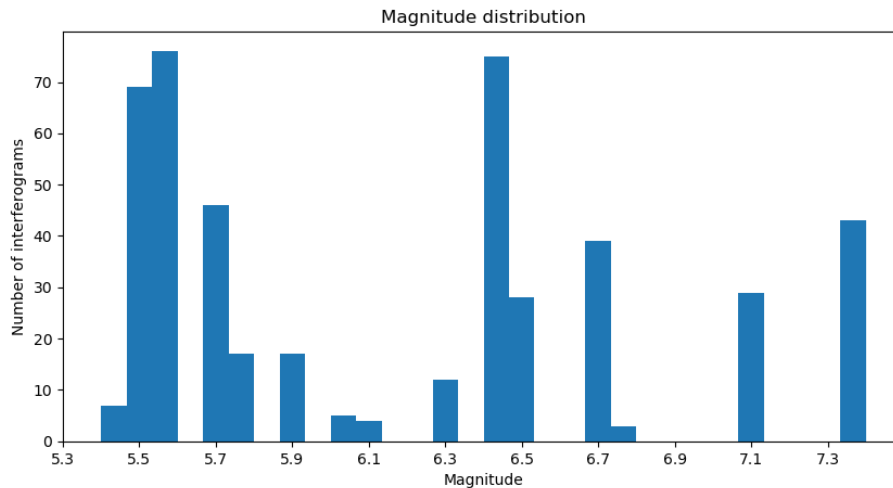


Figure 18: Earthquakes magnitude distribution for downloaded interferograms

Two big groups with more interferograms are visible and can be seen in the earthquakes magnitude distribution histogram (Figure 18):

- The first one is for more minor earthquakes, with fewer fringes between 5.4 and 5.6 of magnitude having more than 140 interferograms.
- The other is a smaller group with a little less than 100 interferograms with a medium magnitude.

Additionally, there are three earthquakes, the largest magnitudes, 7.1 (one earthquake, California) and 7.4 (two earthquakes, Qinghai in China, and Oaxaca in México), 29 and 43 interferograms, respectively.

In this case, depth is not a good sign to analyse because we obtained United States Geological Survey (USGS) data, and when exist insufficient data to a proper value is used 10km as a fixed value. However worth note some earthquakes occurred in shallow depths like the Tonopah, Nevada earthquake. This type of earthquake (in shallow depths) can achieve high dimensions even with low magnitude.

Table 2: Information about selected earthquakes.

Zone	Date	Magnitude	Depth (KM)	Number of interferograms
Álftanes, Iceland	20/10/2020	5.5	10	22
Arzak, China	19/01/2020	6	6	5
Challis, Idaho	31/03/2020	5.6	15	5
Chukotskiy Avtonomnyy Okrug, Russia	09/01/2020	6.4	10	10
Dali, China	21/05/2021	6.1	10	4
Elazig, Turkey	24/01/2020	6.7	10	39
Grindavík, Iceland	24/02/2021	5.6	10	37
Hotan, China	25/06/2020	6.4	10	14
Idgah, Pakistan	30/12/2019	5.4	14	7
Kanallakion, Greece	21/03/2020	5.7	10	14
Karakenja, Tajikistan	24/01/2020	5.5	10	2
Kirkagac, Turkey	22/01/2020	5.6	6	20
Magna, Utah	18/03/2020	5.7	12	8
Mamurras, Albania	26/11/2019	6.4	22	24
Mohr, Iran	09/06/2020	5.5	10	19
Nagqu, China	19/03/2021	5.7	10	11
Oaxaca, México	23/06/2020	7.4	26	14
Petrinja, Croatia	29/12/2020	6.4	10	24
Qinghai, China	21/05/2021	7.4	10	29
Ridgecrest, California	06/07/2019	7.1	8	29
Saray, Turkey	23/02/2020	5.8	10	17
Tallaboa, Puerto Rico	07/01/2020	6.4	9	3
Tonopah, Nevada	15/05/2020	6.5	3	28
Turt, Mongolia	11/01/2021	6.8	10	3
Týrnavos, Greece	04/03/2021	5.5	10	26
Xegar, China	20/03/2020	5.7	10	13
Xizang, China	22/07/2020	6.3	10	12
western Xizang, China	29/03/2021	5.6	10	14
Yedisu, Turkey	14/06/2020	5.9	10	17

Analysing the interferograms, they can be divided into three types: i) the big earthquakes, ii) the medium earthquakes. Iii) the small earthquakes. This division was essential to split the data into sets in the next step to balance the sets as fairly as possible.

As expected, the big earthquakes, with a magnitude of 7.1 and 7.3, cover more extensive areas and have more fringes, as shown in Qinghai, Oaxaca, and California earthquakes. These types of earthquakes are very well visible in both wrapped and unwrapped interferograms. Note that Qinghai, China earthquake is so extensive that the demonstrated interferogram only shows part of him. The rest of the earthquake deformation is on another interferogram. Thus, this specific earthquake covers four interferograms zones been the most extensive of the dataset. The figure below shows some examples of interferograms from extensive surface deformation provoked by earthquakes inside the red boxes (Figure 19).

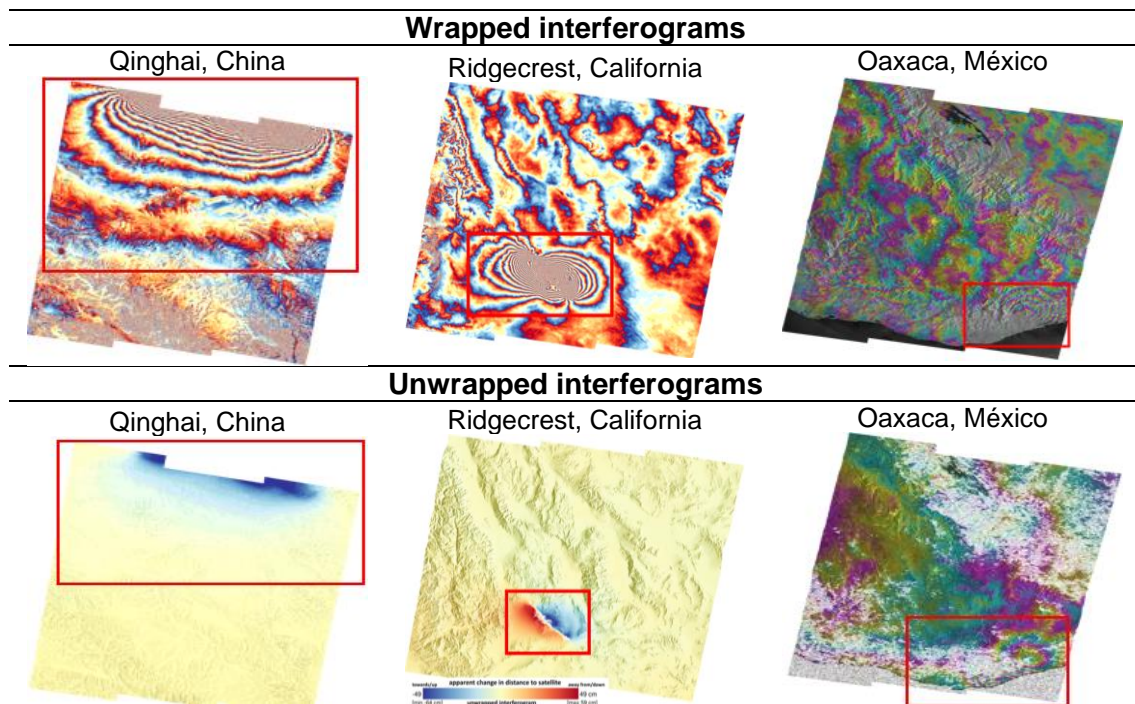


Figure 19: Examples of interferograms from big earthquakes zones

Medium earthquakes are the most common. They can be detected easily in both wrapped and unwrapped interferograms. The fault is well seen in the majority of these earthquakes. However, if the dates of the SAR images are to separate, the fringes furthest from the epicenter start to vanish. These earthquakes go from 5.8 to 6.7 of Magnitude. Can be some exceptions if the earthquake is below 5.8 but occurs in shallow depth or if the magnitude is bigger than 6.7 but occurs at higher depths (Figure 20).

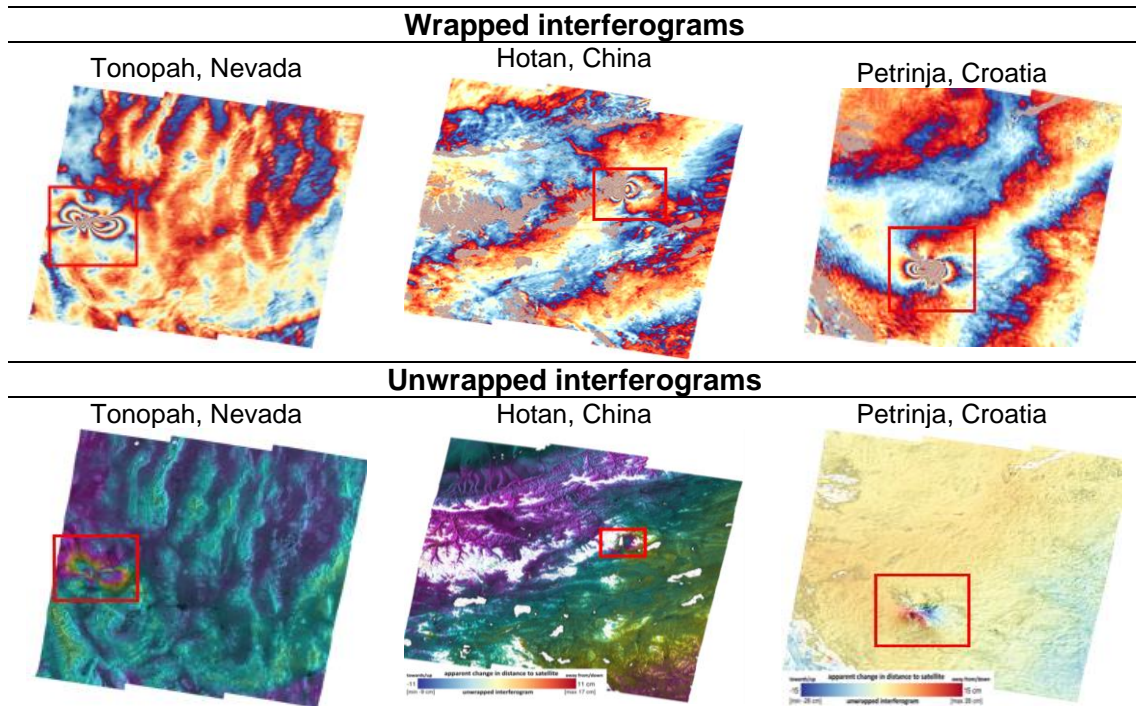


Figure 20: Examples of interferogram from medium earthquakes zones

Small earthquakes are the most difficult to detect. These can be “hidden” in the atmospheric noise. Wrapped interferograms usually have very few fringes (or even only one), and his fault goes from difficult to impossible to see. In the unwrapped interferograms, they are even more challenging to see. His fault is also difficult or impossible to see. Although, the right color scale can help to find him in unwrapped interferograms. The magnitude is usually less than 5.6.

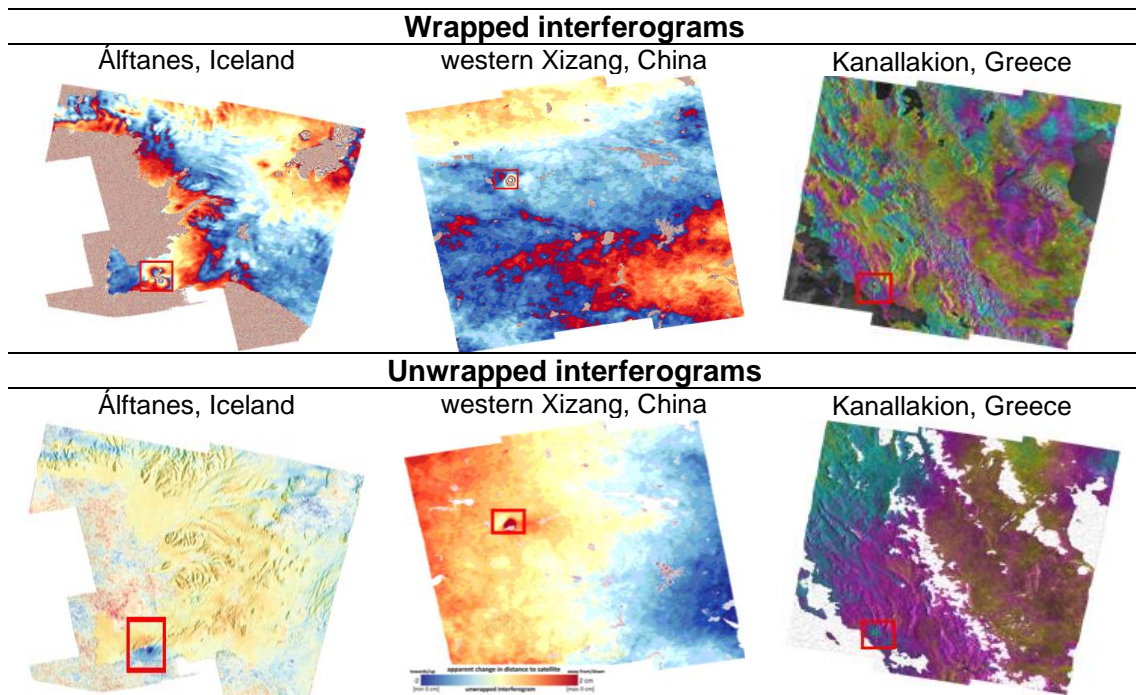


Figure 21: Examples of interferograms from small earthquakes

3.2. Data preparation

Deep learning dimension models require computational resources proportional to the input images to process. The downloaded interferograms represent vast areas and are stored in large images. Though, it was necessary to find patches that could represent earthquakes fringes. Thus, after analysing the interferograms, it was found that patches with 256x256 pixels size is adequate.

To prepare the data for the model's train was necessary to normalize the data. Therefore, the data was prepared according to the following pipeline (Figure 22).

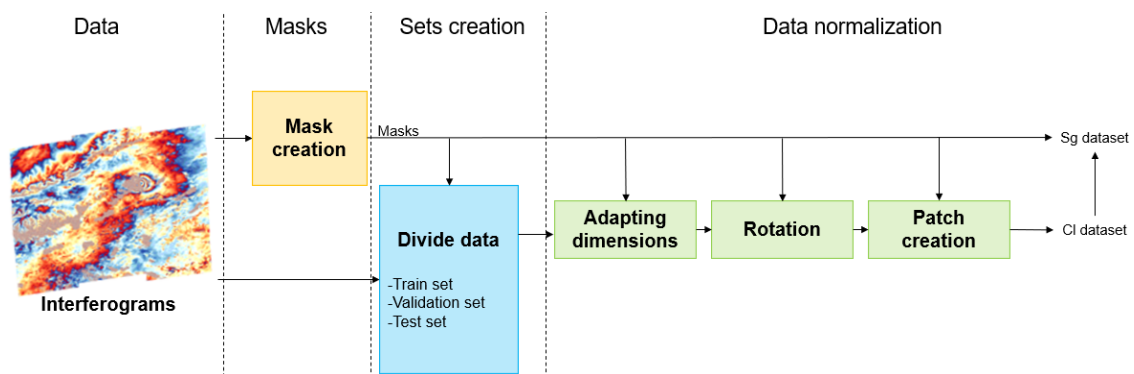


Figure 22: Data preparation pipeline

First, deformation fringes were manually annotated with a mask with an image tool (Adobe Photoshop), as shown in Figure 23.

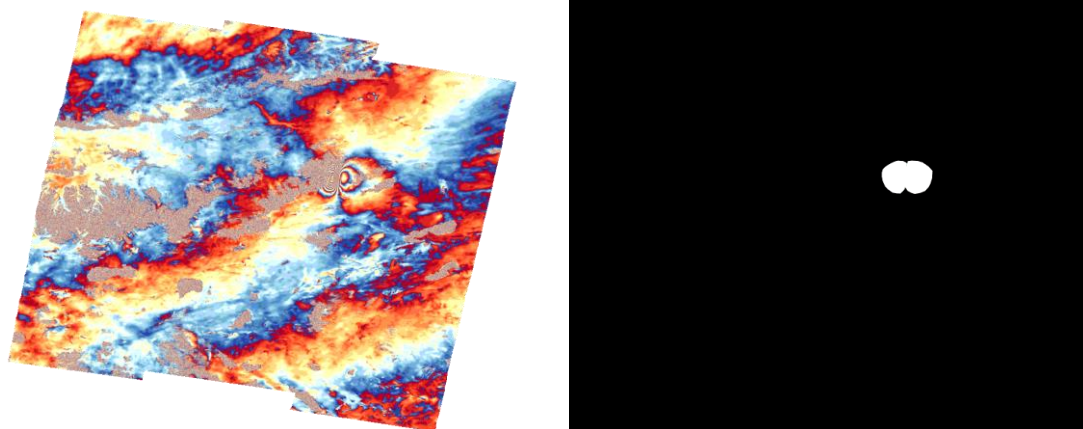


Figure 23: Earthquake interferogram (left) and his mask (right)

The mask has a small margin to test if the segmentation models can find deformation "hidden" in the atmospheric noise. All the steps below apply to the interferogram were also applied to the mask.

All images were initially normalised since machine learning methods, including deep learning, benefits from this operation.

To assure that models are not trained and evaluated with the same earthquakes, interferograms data was distributed by earthquakes into three sets: train set (~60% of the earthquakes), validation set (~20% of the earthquakes), and test set (~20% of the earthquakes). All sets guarantee big and small earthquakes since earthquakes were individually assigned to their set (Table 3).

Table 3: Number of interferograms for each set of each dataset (wrapped and unwrapped)

Set	Deformation	No deformation
Train	254	2189
Validation	94	340
Test	122	262

Interferograms represent the same area size in km, but all the images do not have the exact dimensions. Therefore, to approximate the size of pixels was decided to resemble all interferograms to 1024x1024pixels.

The raw interferograms came with a rotation, as shown in Figure 19, Figure 20, and Figure 21. This rotation can be for the right or left, depending on whether the satellite is ascending or descending orbit. As a result, the interferograms were aligned with the angle of the satellite's passage, eliminating the rotation. Without eliminating the rotation, the patches of the edges had much blank areas (Figure 24).

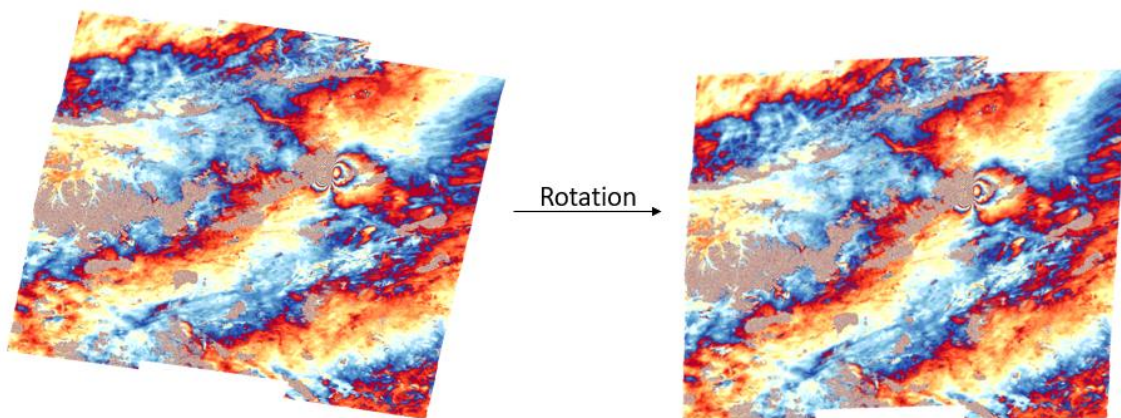


Figure 24: Demonstration of interferogram rotation

Next, we cut the interferograms in patches of 256x256 pixels. Interferograms without deformation fringes were cut into parallel patches. For interferograms with deformation fringes, the patches were overlapped with a stride of 128 horizontally and vertically to guarantee the deformation in the patches center; deep learning models are sensitive to the object's location. Patches where the deformation was not in the 150x150 pixels center were considered null and eliminated (Figure 25).

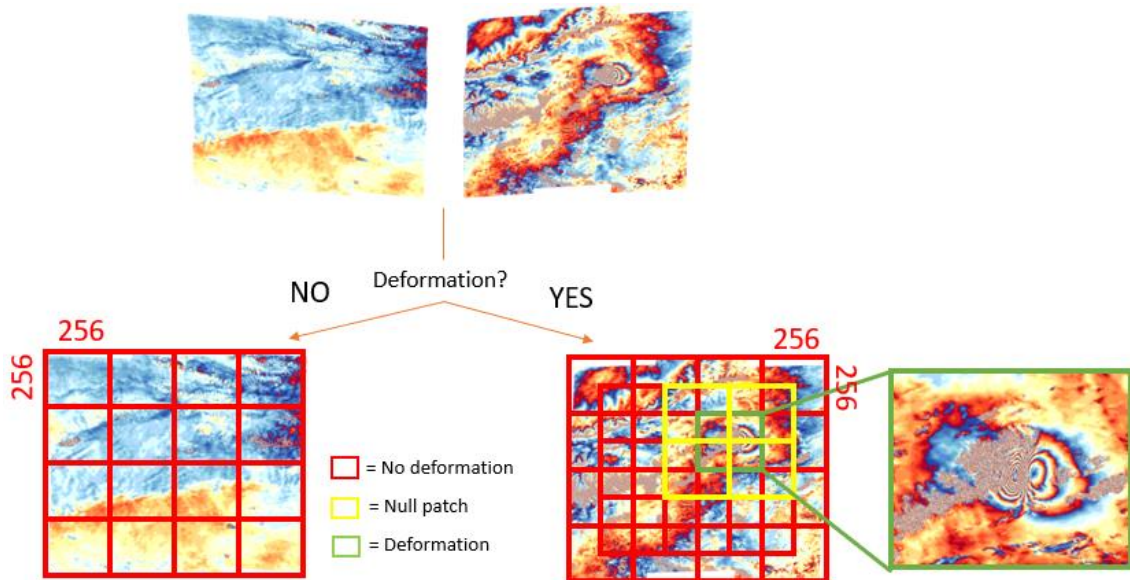


Figure 25: Demonstration of patch creation

As shown in Figure 25:

- The red patches do not have the deformation inside;
- The yellow patches have the deformation. However, the deformation does not belong to the 150x150 pixels patch center;
- The green patch includes the deformation, and the deformation is on the patch center.

Finally, the color channels of the interferograms were normalized to a greyscale. So, we change the pixel's values from $[0, 256]$ to $[0, 1]$ since the machine learning models perform better dealing with small values.

After applying all these steps, the final datasets were for each wrapped and unwrapped interferogram are presented in Table 4.

Table 4: Final Dataset in patches (number of interferogram patches for a set)

Set	Class dataset		Seg dataset
	Deformation	No deformation	Only deformation
Train	499	14979	499
Validation	380	4051	380
Test	253	3826	253

The datasets were very unbalanced. This happened because the number of interferograms without deformation fringes is much more significant than those with deformation fringes. Therefore, there is necessary to use techniques to deal with this unbalanced.

3.3. Models train

After the creation of classification and segmentation datasets, the models were trained. It follows the methodologies followed for deal with the unbalanced data and small datasets and training the models.

3.3.1. Dealing with unbalanced data and small datasets

When datasets are unbalanced, i.e., they have very different amounts of data for each class, models tend to become more difficult to learning correctly. An unbalanced dataset will bias the prediction model towards the more common class (Lee & Lee, 2012). Some techniques deal with unbalanced data, including data augmentation (Mikołajczyk & Grochowski, 2018) and focal loss (Lin et al., 2017). Another common problem in training deep learning models is the shortage of data for which transfer learning is usually used.

Data augmentation is a technique in which data is created for the class with the least amount of data, thus creating synthetic data. There are several ways to do this, but the most common is through traditional transformations, i.e., geometric deformations are applied to the images, which can be, zoom in/out, reflection, shear, rotation, among many others. Figure 26 shows some examples of common data augmentation transformations.

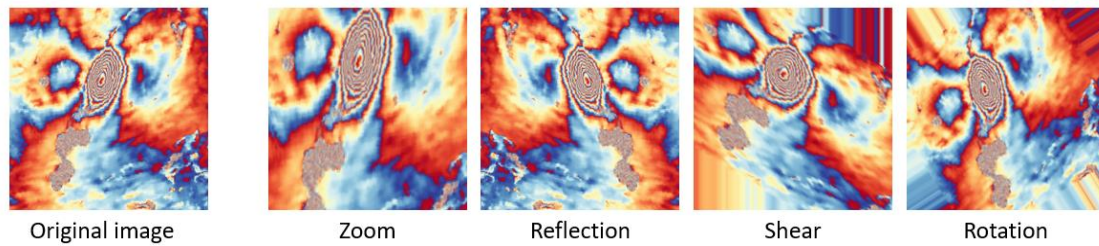


Figure 26: Example of data augmentation

Focal loss is a variant of cross-entropy loss. For the Cross-entropy, if the prediction is identical to the correct answer, the loss will be low. Otherwise, it will be high. Thus, the cross-entropy idea is to penalise wrong predictions more than rewarding correct ones (5). Unfortunately, this brings a problem because after adding all the small losses on the images can overcome the overall loss (total loss). Thus, it leads to degenerate models. Focal Loss changes the weight factors to a factor of $(1 - pt)^\gamma$ to the standard cross-entropy criterion. This way, the bigger the γ , the lowers the loss. Setting $\gamma > 0$ reduces the relative loss for well-classified examples ($pt > 0.5$), putting more focus on hard, misclassified examples:

$$FL(pt) = -\alpha (1 - pt)^\gamma \log(pt) \quad (12)$$

Where:

pt = predicted probability given by the model.

FL = Focal Loss.

γ = values between $[0, 5]$, 0 corresponds to cross-entropy.

α = weightage factor $[0, 1]$.

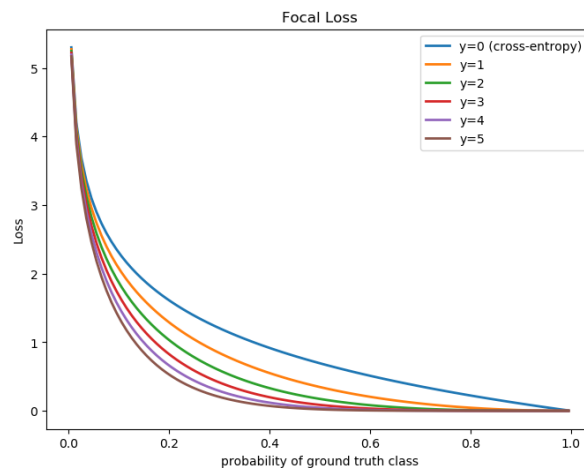


Figure 27: Comparison between cross-entropy and focal loss, adapted figure from T. Lin 2017

Figure 27 shows the evolution chart of the loss for each value of γ . The authors tested Focal loss for the RetinaNet network and achieved the best results with $\alpha=0.25$ and $\gamma=2$.

Transfer learning is a widely used technique for deep learning problems used to deal with small datasets. It consists of using pre-trained models trained in wide datasets and using the weights of these models for our dataset. One way to use transfer learning is through fine-tuning. This technique consists of freezing all the base layers of the pre-trained model and training only the last convolutional layer block and the layers added by the operator, usually the fully connected classifier. When the model starts to overfit its unfreeze all layers and trains the model for a few more epochs obtaining a slight improvement of the model (Hinton et al., 2006).

3.3.2. Classification models

Three state-of-the-art pre-trained models to classify the deformation fringes created by the earthquakes were chosen: InceptionV3, VGG19, and ResNet50V2.

These models were selected because they are an evolution of models that achieve the best accuracy in ImageNet database, and they achieve satisfactory results in preliminary tests.

First, was used the pre-trained models with the ImageNet dataset. Then, the model operates transfer learning with fine-tuning at the top, stratifying the first layers, and substituting and training the final classifier. For the classifier, we use fully connected layers (dense layers interspersed with batch normalisation layer as regularisation) to achieve the output after the convolutional model. The output was a sigmoid, i.e., a value between 0,1 which means the probability of having deformation fringes (Figure 28).

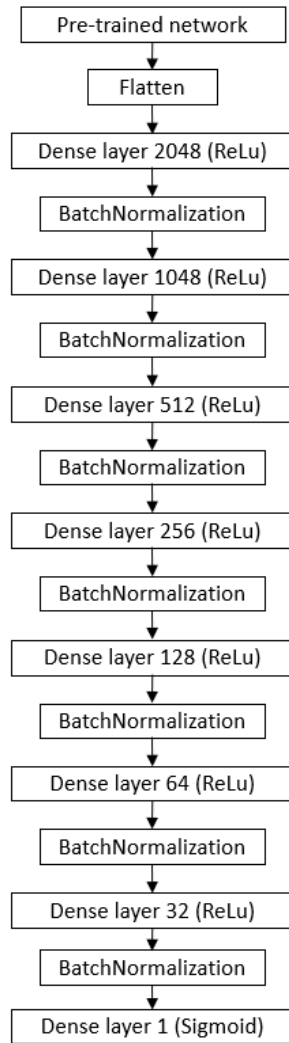


Figure 28: Architecture of the classification models

The train of the classification models started in parallel with the dataset's creation, initially starting with a small dataset with 1800 interferograms until we got the entire dataset of Table 4. This was a smaller and much more controlled dataset (Table 5).

Table 5: Preliminary dataset (number of patches)

Set	Deformation	No deformation
Train	287	5855
Validation	146	1326
Test	166	166

Data augmentation was used in the positive cases for the train and validation set to equilibrate the positive/negative cases to deal with the unbalanced data. However, as

the dataset grew, this technique became useless and needed other options. So then, it was decided to use the focal loss as the loss function instead of the data augmentation.

The model used the Adam optimiser with a learning rate of 0.0001, binary cross-entropy for the loss function when using data augmentation on positive cases, and focal loss when not using data augmentation. To speed up the process, and for computational reasons, data was placed in generators with a 20 batch size. It was decided on 260 steps per epoch and 60 validation steps. The model was trained for 150 epochs. However, the algorithm had a call back that only saves the best model based on the loss score.

3.3.3. Segmentation models

To isolate the area affected by the earthquake were chosen two segmentation models, Dual attention and Unet. For the Unet model, we evaluate the three convolutional neural networks trained in the classification (VGG19, InceptionV2, ResNet50V2) as the encoder. As explained before, Unet is composed of an encoder and a decoder, and Dual attention combines two segmentation techniques (position attention and channel attention). Although Unet was chosen as the most used model for this purpose, Dual attention is a recent technique that can achieve good results.

The model was trained for 20 epochs with the callback that only saves the best model. After the 20 epochs, the model starts to overfitting. The model uses the Adam optimiser with a learning rate of 0.001 and a Jaccard coefficient as the loss function (IoU). Finally, the data were put in generators of 20 batch sizes again and chose 125 steps per epoch and 62 validation steps.

The segmentation models only used the positive cases of the dataset. When tested with both positive and negative the results were poor. Therefore, the segmentation model cannot distinguish patches with deformation and patches without deformation but only detect the deformation in patches with deformation. Consequently, the classification model must be used to detect the patches with deformation before using the segmentation model for practical use.

3.4. Model evaluation

The convention set metrics to evaluate classification and segmentation models were used. For classification, were used accuracy, precision, recall, F1 score (Goutte & Gaussier, 2005), G-mean (Guo et al., 2016), ROC Curve, Area Under the Curve (AUC),

and for segmentation were used IoU Score and Dice coefficient. In the following sections, we present metrics definitions.

3.4.1. Classification metrics

The ground-truth value and the model prediction are used to evaluate classification models. Models' predictions are classified as:

- True Positive (TP), when the prediction is positive and equal to the ground-truth;
- True Negative (TN), when the prediction is negative and equal to the ground-truth;
- False Positive (FP), when the prediction is positive, and the ground-truth is negative;
- False Negative (FN), when the prediction is negative, and the ground-truth is positive.

The total number of each TP, TN FP and FN are the basic metrics. These metrics are frequently summarised in the confusion matrix, as represented in Figure 29.

		True class	
		Positive	Negative
Predictive class	Positive	TP (True Positive)	FP (False Positive)
	Negative	FN (False Negative)	TN (True Negative)

Figure 29: Confusion Matrix

Accuracy is the general metric and is defined by the percentage of correct predictions in all population test.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (13)$$

Precision is the percentage of correct answers among all answers given as positive by the model. Low precision means that exist an high rate of false positives

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

The recall is the percentage of correct answers out of all expected positive answers. Low recall means that exist an high rate of false positives

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

G-mean is a metric used to find the best threshold; it is the geometric mean of recall and precision

$$Gmean = \sqrt{Recall * Precision} \tag{16}$$

F1 score is a harmonic mean between precision and recall.

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{17}$$

The Receiver Operating Characteristic curve (Hoo et al., 2017) or ROC curve consists of the plot of the true positive and false positive rates calculated for each threshold, as shown in Figure 30.

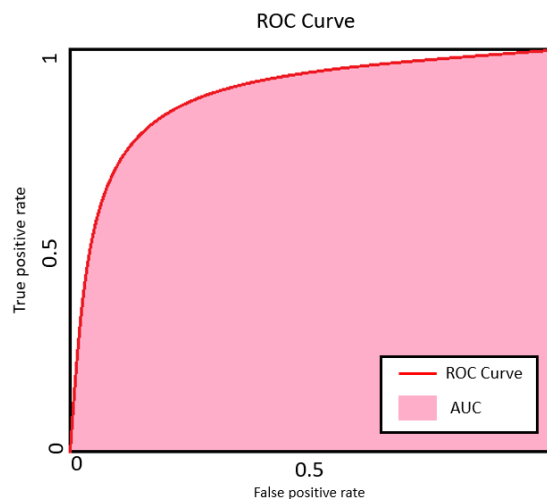
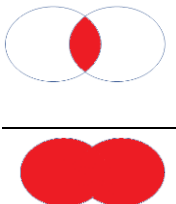


Figure 30: Example of a ROC Curve

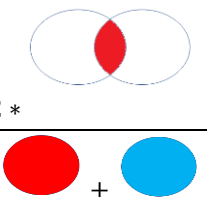
This graph permits defining the best threshold for the model balancing the true positive rate with the false positive rate. The area under the ROC Curve (AUC) is the percentage of the metric score that characterises the model independently of the selected thresholds - a graph under the ROC Curve (Figure 30). Thus, as bigger the AUC is near 1, it usually means the better is the model.

3.4.2. Segmentation metrics

IoU (Intersection Over Union) score (Nowozin, 2014) is a metric that evaluates how similar are two areas, the predicted area, and the ground truth area. They can be calculated from the evaluation of each image pixel as TP, TN, FN, and FP, according to equation 18.

$$IoU\ score = \frac{TP}{TP + FN + FP} = \frac{Area\ of\ overlap}{Area\ of\ union} = \frac{\text{Diagram}}{\text{Diagram}} \quad (18)$$


Dice coefficient (Fidon et al., 2018) is another widely used metric to evaluate how similar two areas are and is defined by the following formula.

$$Dice = \frac{2 * TP}{2 * TP + FN + FP} = \frac{2 * Area\ of\ overlap}{Area\ predicted + Ground\ truth} = \frac{2 * \text{Diagram}}{\text{Diagram} + \text{Diagram}} \quad (19)$$


4. Results and discussion

In this chapter, the results of the different fringes detection and segmentation methods are presented, discussed, and compared to the published related works. First, in section 4.1, we present the results of the automatic classification approaches; followed, in section 4.2, by the results of the earthquakes localisation based on the segmentation methods.

4.1. Classification

Various tests were made with different setups to train the models InceptionV3, VGG19, and ResNet50V2. This section shows the most important results achieved: i) the preliminary results with the smaller dataset, ii) the results achieved with data augmentation in the final dataset, iii) the preliminary models applied to the final test set and, iv) the final dataset with the loss function instead of the data augmentation.

Preliminary results

The preliminary test with the smaller dataset (Table 5) and data augmentation to balance the dataset achieved the following results. Note that the test set was balanced, but data augmentation was not used.

Table 6: Metrics for the preliminary tests. The threshold at 50%.

Wrapped interferograms			
Model	Accuracy	F1-Score	AUC*
InceptionV3	0.84	0.85	0.90
VGG19	0.89	0.87	0.96
ResNet50V2	0.80	0.80	0.87
Unwrapped interferograms			
Model	Accuracy	F1-Score	AUC*
InceptionV3	0.68	0.69	0.74
VGG19	0.71	0.74	0.77
ResNet50V2	0.70	0.75	0.77

*value not affected for the threshold

For the preliminary tests, the results obtained were great and showed deep learning capability in fringes detection, mainly for wrapped interferograms (Table 6). The

model VGG19 showed to be slightly better than the other models on wrapped interferograms, achieving an accuracy of 0.89 against 0.84 and 0.80 for the InceptionV3 and ResNet50v2 models. VGG19 also obtained a better F1-Score and AUC than the other models.

On unwrapped interferograms, the models VGG19 and ResNet50V2 achieved identical results with 0.71 and 0.70 for accuracy, 0.74 and 0.75 for F1-Score, and 0.77 for the AUC. The InceptionV3 performs slightly worse than the other models.

The metrics show that all models perform better on the wrapped interferograms.

The ROC curves confirm these affirmations with a cleaner view and show the performance regardless of the threshold (Figure 31).

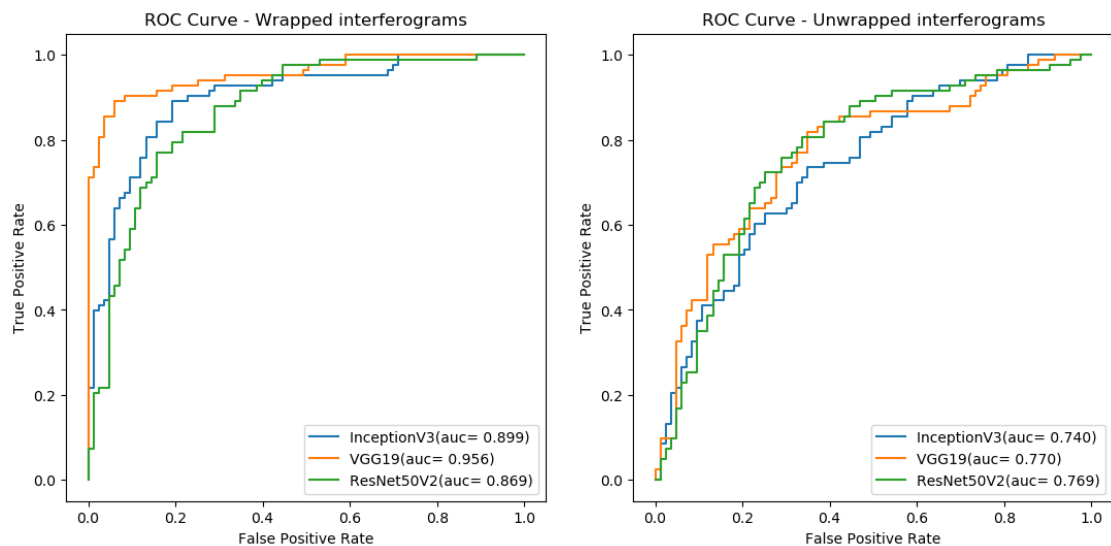


Figure 31: ROC Curves for the preliminary tests. At left wrapped interferograms, at right, unwrapped interferograms

Analysing the ROC curves, VGG19 for wrapped interferograms keeps showing to be the better choice. Wrapped interferograms are better than unwrapped to use deep learning techniques regardless of the threshold. For the unwrapped interferograms, all models perform identically

Data augmentation with the final dataset

After the promising results achieved in the preliminary tests, the same techniques and setups were used in the final dataset (Table 4) achieved the following results (Table 7). More metrics will be used after this point because the test set is unbalanced and more metrics are needed to better understand the results and the models.

Table 7: Metrics for data augmentation applied on the final dataset—the threshold at 50%.

Wrapped interferograms					
Model	Accuracy	Precision	Recall	F1-Score	AUC*
InceptionV3	0.63	0.08	0.54	0.15	0.64
VGG19	0.80	0.14	0.41	0.21	0.70
ResNet50V2	0.63	0.08	0.51	0.14	0.59
Unwrapped interferograms					
Model	Accuracy	Precision	Recall	F1-Score	AUC*
InceptionV3	0.49	0.07	0.62	0.13	0.57
VGG19	0.26	0.06	0.83	0.12	0.51
ResNet50V2	0.62	0.08	0.49	0.14	0.58

*Not affected for the threshold

These results do not corroborate the good results achieved in the preliminary tests. These models achieved a maximum of 0.21 for the F1-Score with the VGG19 model for the wrapped interferograms. The results are even worse for the unwrapped interferograms, reaching a maximum of 0.14 F1-Score for ResNet50V2. The shallow values for the precision for all models show that the models have a high rate of False Positives, showing the incapability of the model to detect the negative cases. This low precision also happened because the data is unbalanced, having many more negative cases than positive; this is another reason for the low precision values. Even though the results slightly improved adapting the threshold, that is not enough to achieve satisfactory results, as the ROC curves and AUC are both low (Figure 32).

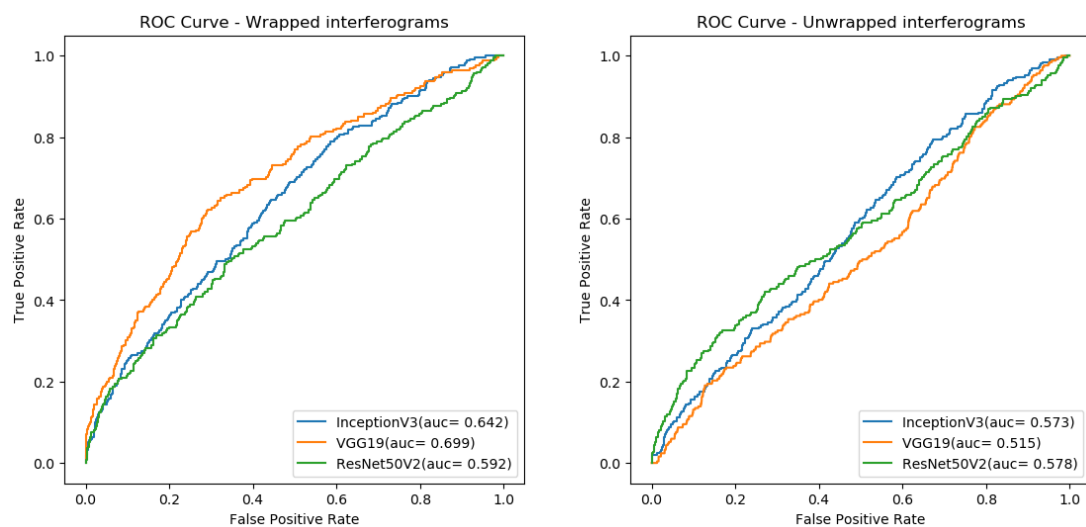


Figure 32: ROC Curves for models using data augmentation on the final dataset. At left wrapped interferograms, at right, unwrapped interferograms

For the unwrapped interferograms with the AUC values near 0.5, that means that the model's predictions are near random. The 0.49 of accuracy on the InceptionV3 model for unwrapped interferogram means the same, even more, when the dataset is unbalanced. The 0.29 accuracy on the VGG19 was a strange value. However, the ROC curves and AUC (0.52) show that the values, even being lower than the other models (0.57 and 0.58), can be comparable, which means that adapting the threshold the accuracy will be better despite that all these values achieve in this models are not satisfactory.

Preliminary dataset models on the final test set

Due to the results achieved using data augmentation in the final dataset, the preliminary models were used on the final test set for a fair comparison. This way is possible to conclude if the problem is the models or the data (Table 8).

Table 8: Metrics for preliminary models applied on the final dataset. The threshold at 50%.

Wrapped interferograms					
Model	Accuracy	Precision	Recall	F1-Score	AUC*
InceptionV3	0.69	0.12	0.63	0.20	0.70
VGG19	0.80	0.17	0.60	0.25	0.78
ResNet50V2	0.46	0.11	0.65	0.19	0.72
Unwrapped interferograms					
Model	Accuracy	Precision	Recall	F1-Score	AUC*
InceptionV3	0.60	0.09	0.60	0.16	0.66
VGG19	0.50	0.07	0.63	0.13	0.56
ResNet50V2	0.43	0.08	0.82	0.15	0.68

*Not affected for the threshold

As expected, these results were worse than the ones obtained on the preliminary tests. This happened because this dataset is more extensive and unbalanced despite the models being the same. However, the most unexpected is that the preliminary models results were better than the final ones with data augmentation when applied to the same tests, even being trained with fewer data.

Comparing the models for the 50% threshold, only the model ResNet50V2 performs identically in both datasets. The models trained in the final dataset (Table 7) achieved better accuracy and precision, this one achieved better recall and F1-Score. For all the other models, the preliminary models (these ones) achieved better results.

This difference gets more notorious on the ROC Curves and AUC (Figure 33).

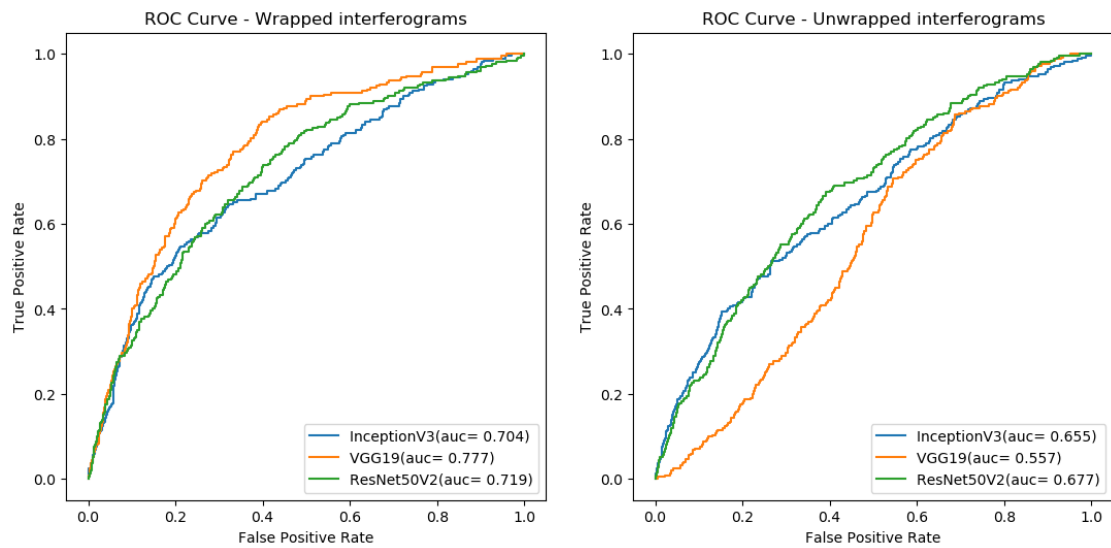


Figure 33: ROC Curves of the preliminary models applied on the final test set. At left wrapped interferograms, at right, unwrapped interferograms

Analysing the ROC Curves and AUC independent of the threshold, the difference is more notable mainly for wrapped interferograms. All models have better performance than the final models presented in Figure 32 with an AUC of 0.70, 0.78, and 0.72 against 0.64, 0.70, and 0.60.

The same happened on unwrapped interferograms but with a minor difference with an AUC of 0.63, 0.56, and 0.68 against 0.54, 0.51, and 0.58. Note that the more identical model (ResNet50V2) for the 50% threshold has a more significant difference on the AUC (0.12 for wrapped and 0.10 for unwrapped interferograms).

After these results are possible, conclude that data augmentation is good up to a point, but as the imbalance gets bigger, the quality of the model starts to fail. Thus, applying a new method that outperforms these models was necessary since those not achieved satisfactory results.

Focal Loss on final dataset

The focal loss was tried instead of data augmentation. As shown before, this is a particular type of loss that helps to lead to unbalanced data. The results can be seen in Table 9.

Table 9: Metrics achieved for wrapped and unwrapped interferograms patches. The threshold at 50%

Wrapped interferograms					
Model	Accuracy	Precision	Recall	F1-Score	AUC*
InceptionV3	0.95	0.95	0.23	0.37	0.77
VGG19	0.97	1.00	0.52	0.69	0.86
ResNet50V2	0.96	1.00	0.36	0.52	0.75
Unwrapped interferograms					
Model	Accuracy	Precision	Recall	F1-Score	AUC*
InceptionV3	0.94	0.53	0.08	0.13	0.65
VGG19	0.95	0.88	0.20	0.32	0.73
ResNet50V2	0.94	1.00	0.08	0.15	0.67

* not affected by the threshold

The accuracy metric is not fair in these cases (they are all very high between 0.952 and 0.970) because it is an unbalanced dataset. However, these high values are already indicative of improvement comparing with the other models shown above.

The most unexpected was the value of precision. Once that the data without deformation fringes is much more extensive than the ones with them is expecting that the precision achieves low values due to the amount of false positives but that doesn't happen, reaching values of one or near it. The exceptions were the Inception V3 and VGG19 for unwrapping interferograms (0.53 and 0.87, also good results). Not finding false positives is an excellent result because the models were tested in more than 3800 negative cases. The threshold can be lowered, and the metrics will probably improve. The right threshold depends on the user and how much FP/FN is suitable for the specific case.

The values of the Recall were low. Therefore, without data augmentation, exist fewer positive cases to train and "teach" the model achieving more false negatives on the tests. However, the 0.52 achieved by the VGG19 for wrapped interferogram are exciting results, even more, when the F1-Score is 0.69.

The F1-Score characterises the best performance of the models since it makes a consensus between these two metrics (equation 17), separating the best performance of the models (values 0.370, 0.52, and 0.69) and thus allows to complete that the VGG19 model will present the best performance.

Analysing the ROC curves and AUC is also concluded the same, regardless of the threshold decided (Figure 34).

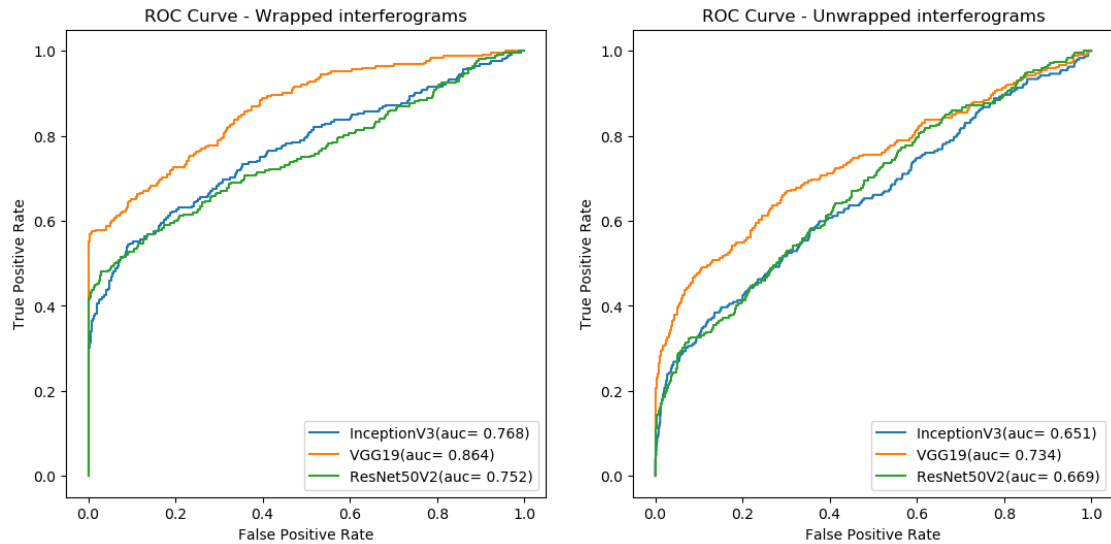


Figure 34: ROC Curves for the classification models using focal loss: at left the ROC curve for wrapped interferograms, and at right the ROC curves for unwrapped interferograms.

As expected, wrapped interferograms allowed better results compared to unwrapped interferograms. Thus, VGG19 for wrapped interferograms became the best model to detect earthquake fringes, as seen in the ROC Curve and AUC values, not only on these ROC curves but in all of them. InceptionV3 and ResNet50V2 perform identically. Although for precision, recall, and F1Score, we evaluate the model to a 50% threshold, that value can be adapted for some objective, depending on how many false negatives we are willing to have. For example, calculating the g-mean for all thresholds in VGG19 for the train set shows that the general best is 36%, with a g-mean of 0.98; 50% threshold only have 0.92 g-mean.

Focal loss performs better than data augmentation to deal with unbalanced data, as shown in Figure 35. We compare the best model using Focal Loss with the best model using data augmentation.

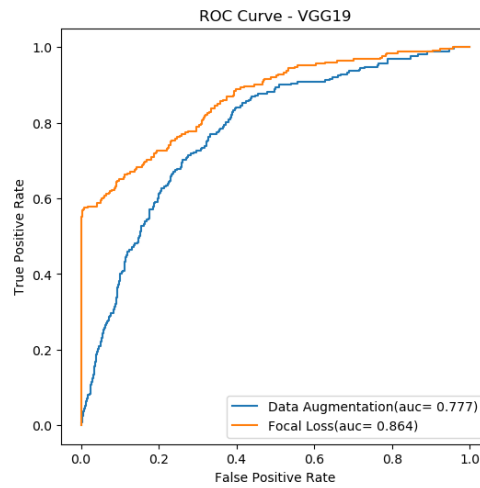


Figure 35: ROC Curve, Focal Loss vs. Data augmentation

Overall, these results can be considered reasonable. However, in practice, the interferograms are cut into overlapped patches. As a result, the model did not detect the earthquake fringes in some patches, that change in the overlapped patch on the same interferogram, as shown in Figure 36.

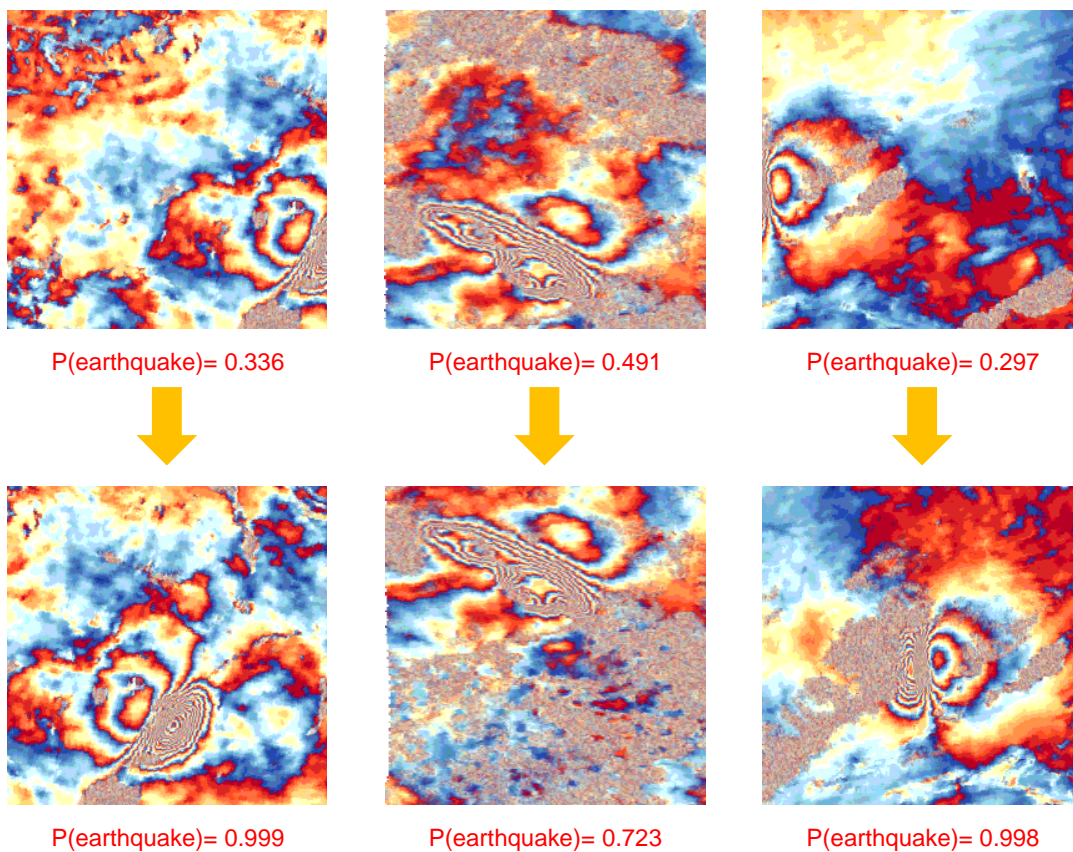


Figure 36: Overlapped earthquakes patches and its probability of having deformation earthquake according to the model

Changing how the positive patches are in data preparation, we can eliminate at first the patches that the models do not found fringes. So, that way, having a better train,

achieving better results. However, considering that for almost all earthquakes are available more than one interferogram, we can guarantee a well-detected positive patch in all cases.

The main objective is found the deformation fringes in interferograms and not in patches. So, it was decided to apply the best models (with focal loss) to calculate the metrics for interferograms and not only for the patches (dataset of Table 3). Thus, the data have a positive prediction (predicted as deformation) if at least one patch of that interferogram is predicted as positive.

The results can be seen below in Table 10.

Table 10: Metrics for tests on full interferograms

Wrapped interferograms				
Model	Accuracy	Precision	Recall	F1-Score
InceptionV3	0.82	0.96	0.44	0.61
VGG19	0.89	1.00	0.64	0.78
ResNet50V2	0.86	1.00	0.57	0.72
Unwrapped interferograms				
Model	Accuracy	Precision	Recall	F1-Score
InceptionV3	0.72	1.00	0.11	0.21
VGG19	0.69	0.70	0.06	0.11
ResNet50V2	0.69	1.00	0.02	0.04

For this case, the dataset is not so much unbalanced as the other dataset (122 positive cases for 266 negative cases). Thus, accuracy does not need to be discarded.

Like in all other cases, wrapped interferograms have better performance than the unwrapped ones, and it can be now stated that they are indisputably better to be applied with deep learning. The unwrapped interferograms for all cases do not show be a good option, not having any advantage to use.

For wrapped interferograms, VGG19 has better performance in all metrics again. In general, all the metrics were improved compared with the patches dataset. This improvement corroborates the idea that some false negatives are corrected in the overlapped patch.

We believe that it is possible to improve these results significantly to a level that, in the future, will be possible to create systems to monitor the Earth in real-time. For

this, is necessary a more extensive dataset and less unbalanced data collecting more positive cases. However, there is no need to train more negative cases since the results show that these models can already detect the patches without deformations.

4.2. Segmentation

Below is presented the IoU, Dice Score, and Accuracy for Unet and Dual attention segmentation models. As for classification, these metrics are calculated for a 50% threshold (Table 11). Only wrapped interferograms were tested once that they already showed the incapability to be detected.

Table 11: Segmentation Metrics

Model	Coder	IoU Score*	Dice Score*	Accuracy*
Unet	InceptionV3	0.43	0.59	0.83
	VGG19	0.32	0.47	0.85
	Resnet50V2	0.13	0.22	0.80
Dual attention	ResNet50	0.25	0.39	0.84

Note: all metrics were calculated with a threshold of 50%.

As happened for the classification, accuracy is not a good metric and should be ignored, being represented as indicative, as it depends on the area's dimensions to segment. The results were poor, tried several approaches, and needed more data to train the segment models properly. Dual attention does not prove to be a good model to segment earthquake fringes in interferograms.

InceptionV3 obtained the best values in segmentation, with VGG19 reaching values close to InceptionV3. However, InceptionV3 has difficulty detecting the edges between deformation/ no deformation, which is the main factor contributing to some errors in this model. This effect can be seen in Figure 37a. Regarding VGG19, it shows a better capability to find the edges. However, the primary constraint is that VGG 19 interprets some image noise as deformation fringes (Figure 37b).

The models Unet with the Resnet50V2 and Dual attention were poor, unable to locate the fringes on the patch (Figure 37c,d).

Note that for the best models (InceptionV3 and VGG19), the model has more confidence in the center than in the borders.

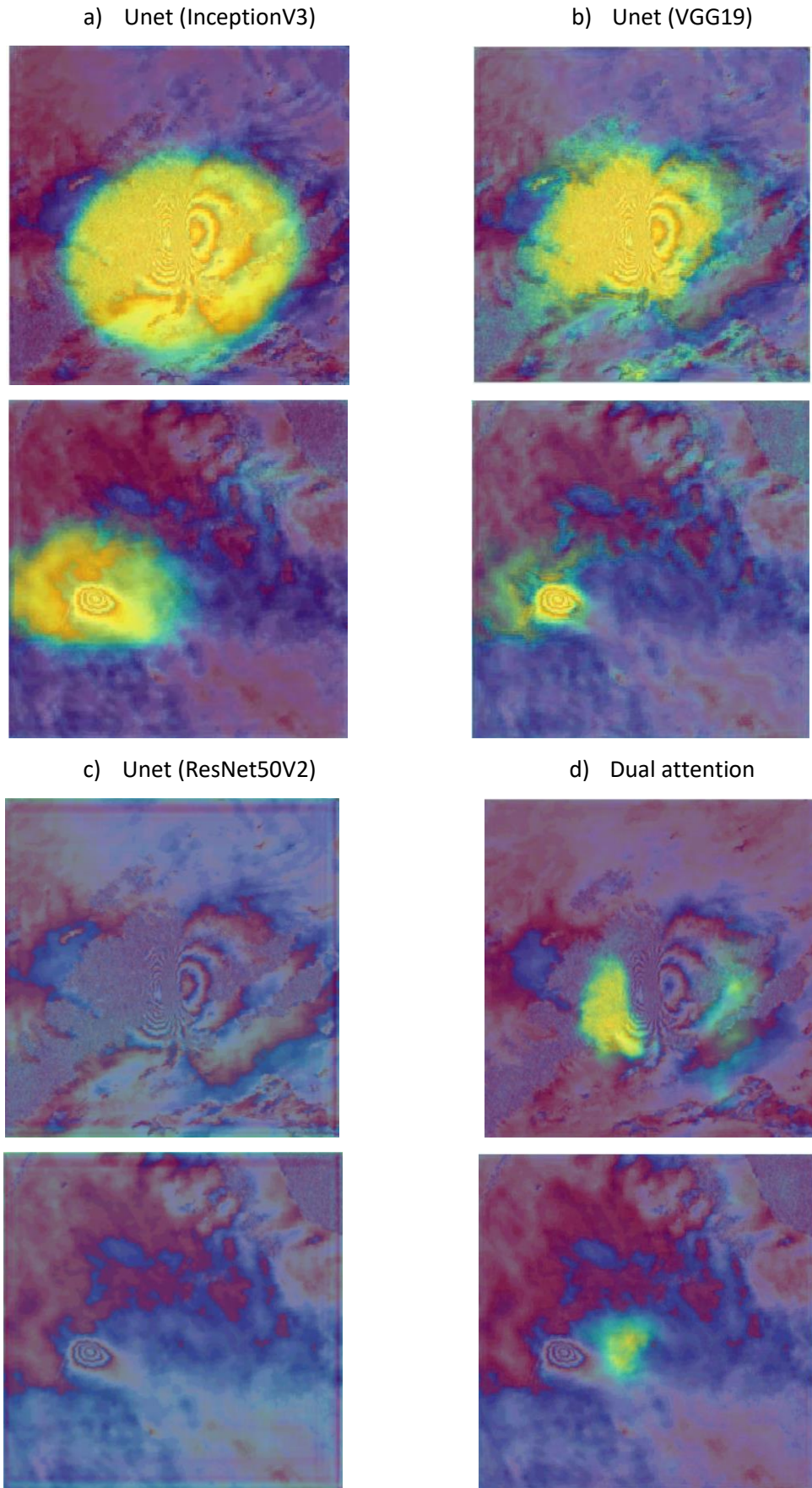


Figure 37: Segmentation confidence area for two earthquakes using a) InceptionV3, b)VGG19, c)ResNet50V2, and d) Dual attention

Testing the metrics for all thresholds in the train set, Inception V3 shows a 90% threshold as his best, while the best threshold for VGG19 is 70%. Thus, evaluating the models for the best thresholds, we have an improvement in the metrics.

Table 12: Metrics of segmentation for the best thresholds (90% for InceptionV3 and 70% VGG19)

Unet Models	IoU Score	Dice Score	Accuarcy
InceptionV3	0.48	0.63	0.87
VGG19	0.47	0.62	0.85

These thresholds slighting improve the detection of the deformation fringes, as shown in Figure 38.

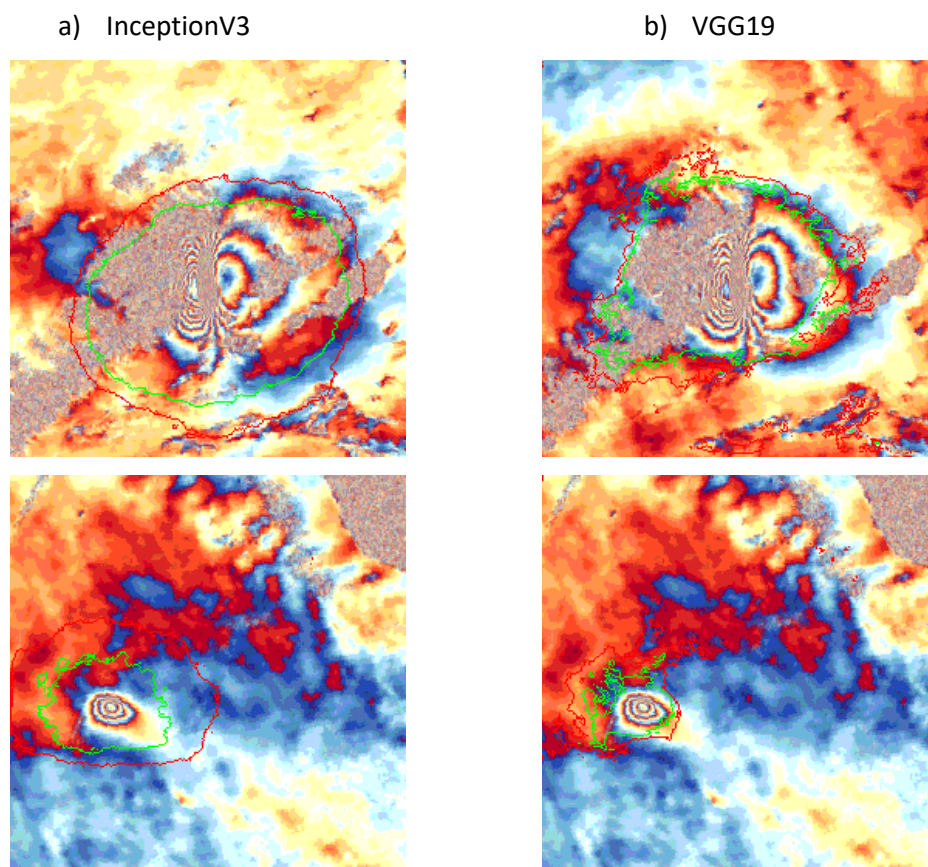


Figure 38: Segmentation borders for two earthquakes using a,b) InceptionV3, c,d)VGG19; -50%threshold, -best threshold

Using the right threshold, the problems encountered above (Figure 37) are partially resolved, mainly for VGG19, where the noise rarely is detected as an earthquake using the suitable threshold. That is the reason why in Table 12, the VGG19 practically reached the values of InceptionV3.

Here are some examples where the models don't work so well, thus justifying the poor results.

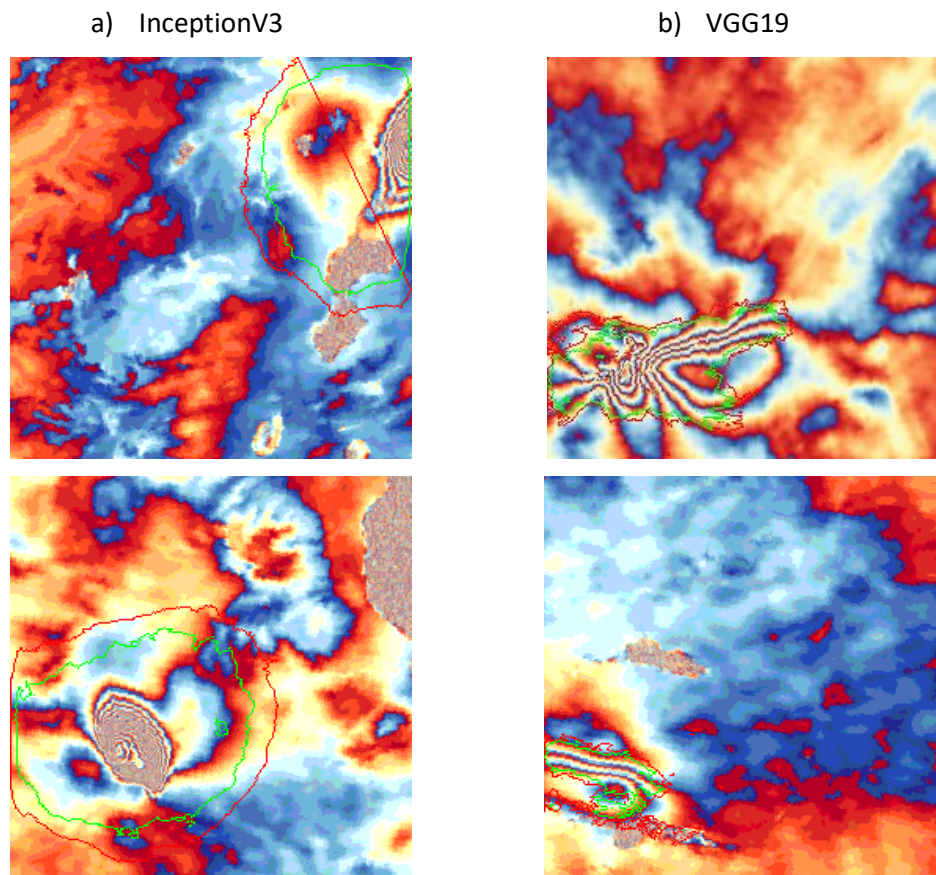


Figure 39: Bad segmentation borders for two earthquakes using a) InceptionV3, b) VGG19; - 50%threshold, -best threshold

These are promising results. However, more data is needed to achieve the desired results. In the future, with more data could be possible to train these models to segment both positive and negative cases.

5. Conclusion

In this dissertation, Machine Learning techniques were explored for automatic detection of seismic deformations in SAR interferograms and estimate the earthquake's impact area - datasets were created, and several models were trained for the classification and segmentation of seismic deformations.

Initially, the literature review was done, where the fundamentals of InSAR, Machine Learning, and Deep Learning were studied and identified the state-of-the-art Deep Learning methods applied to InSAR data. A methodology was devised to address the earthquake fringes classification and segmentation in InSAR tasks, taking into consideration the knowledge learned from the literature review: Sentinel-1 interferograms (wrapped and unwrapped) have adequate information for studying seismic deformations; Deep Learning methodologies are adequate for detecting the deformation fringes, and although small annotated datasets are challenging, it is possible to train Deep Learning models successfully.

Two InSAR datasets were created for training classification models for wrapped and unwrapped interferograms. The data were normalized and manipulated to fit models. However, the available data were too unbalanced due to the small size of deformation areas in the interferograms compared with the areas without deformations, and most of the interferograms do not have any deformations at all. To deal with this, it was exploring the use of "data augmentation" approach and a special loss untitled "focal loss". In a preliminary evaluation, the models were trained with a smaller dataset (during annotation) and with a balanced test set using "data augmentation" (on train and validation) and achieved as best 0.87 and 0.96 for F1-Score and AUC, respectively. When the dataset was complete, the dataset became bigger and more unbalanced. The test set became 15 times more negative cases than positive, and the final evaluation showed poor results with his best an F1-Score of 0.21 and an AUC of 0.70. To enable comparison, we evaluated the models trained in the preliminary dataset on the final test set and achieved were slightly better results - F1-Score of 0.25 and an AUC of 0.78. In this case, "data augmentation" performed worse with a more extensive and more unbalanced dataset than a smaller one. We can conclude that in this case, this method has worse performance when the unbalancing is bigger. We decide to explore the use of "focal loss to overcome this difficulty. This new approach achieved our best results - F1-Score of 0.69 and an AUC of 0.86. Once for each interferogram, we created several overlapped patches. Studying the prediction of overlapped patches, this model does not

show any false positives. For false negatives, in most of the cases overlap true positives. Considering the interferogram has seismic deformation where at least one of the overlapped patches has a positive prediction, the model reached an F1-Score of 0.78, which is a promising result. For all the tests made, VGG19 achieved better results, and the wrapped interferograms consistently achieved better results than the unwrapped ones for all models.

Segmentation models were used in patches of wrapped interferograms that have seismic deformations. Two models were capable of segmenting the earthquake fringes, Unet with InceptionV3 encoder and Unet with VGG19 encoder, achieving a coefficient Dice of 0.59 and 0.47, respectively. We explore other models such as Dual attention and Unet with ResNet50V2 encoder. However, they performed poorly, achieving a coefficient Dice of 0.39 and 0.22, respectively. The best results were achieved with Unet with InceptionV3 encoder (Dice of 0.63) and Unet with VGG19 encoder (Dice of 0.62).

These results corroborate the idea that these Deep Learning methods have the potential to have a faster response to detecting and localising earthquake deformation. We believe that in the future, these results can be improved with a deeper application of deep learning methods. After achieving better results in the future, these methods can be used to detect small deformations with a seismic activity that can proceed a big earthquake giving time to the civilization to be prepared.

The work carried out in the scope of this dissertation, allowed to answer to the 4 research questions:

Question 1: *Are the seismic deformations equally detected in wrapped and unwrapped interferograms?*

Answer 1: No, after doing various tests with different setups and datasets, they all achieved better results in wrapped interferograms than in unwrapped interferograms. This was expected once that wrapped interferograms are a more preliminary version of interferograms containing more information, and being easy to observe.

Question 2: *What is the best state-of-the-art Deep Learning models to detect seismic deformations in SAR interferograms and estimate the earthquake's impact area?*

Answer 2: In this work, three state-of-the-art models to detect seismic deformation and two models with four encoders (3+1) to estimate the earthquake impact area were tested. Among them, VGG19 obtained the best results in almost all cases (including the best result) to detect seismic deformation). To estimate the impact area, Unet with

InceptionV3 as encoder and Unet with VGG19 as encoder obtained similar results, being both the best one to segment the areas.

Question 3: *Although vast public data are available on the COMET website, dataset creation requires a long time and effort to annotate. Thus, with the small dataset possible to create during this work, how efficiently is it possible to train Deep Learning models?*

Answer 3: With our small dataset (mainly positive cases) for classification, using transfer learning, the results were satisfactory and already showed a great capacity of deep learning to detect the deformation fringes in wrapped interferograms. For segmentation, the results were promising, and we believe that more data is needed for improving results.

Question 4: *Available data is very unbalanced – there are a lot of areas without fringes and few with fringes. How to deal with this unbalanced and improve models training?*

Answer 4: This problem appeared in Classification models. Two techniques were tested to deal with the unbalanced data: data augmentation and focal loss. Among them, focal loss outperforms by far the results of data augmentation. Mainly for negative cases, where was not found any false positive for the best-trained model.

For future works, the dataset can be expanded to have more positive data and thus train the models more efficiently. For the segmentation, more models, or combination of models (for example, Unet with InceptionV3 and VGG19) can be tested. After achieving the desirable results on detecting seismic deformation, we believe that the next step should be try to find small deformations that could occur before the event.

References

Curlander, J., and R. McDonough, 1991. Synthetic aperture radar systems and signal processing, New York: John Wiley & Sons.

Lecun, Y., L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, DOI: 10.1109/5.726791.

Agarap, A. F. (2019). *Deep Learning using Rectified Linear Units (ReLU)*. 1, 2–8.
<http://arxiv.org/abs/1803.08375>

Anantrasirichai, N., Achim, A., Morgan, J. E., Erchova, I., & Nicholson, L. (2013). SVM-based texture classification in Optical Coherence Tomography. *Proceedings - International Symposium on Biomedical Imaging*, 1332–1335.
<https://doi.org/10.1109/ISBI.2013.6556778>

Anantrasirichai, N., Biggs, J., Albino, F., & Bull, D. (2019). A deep learning approach to detecting volcano deformation from satellite imagery using synthetic datasets. *Remote Sensing of Environment*, 230(May), 111179.
<https://doi.org/10.1016/j.rse.2019.04.032>

Anantrasirichai, N., Biggs, J., Albino, F., Hill, P., & Bull, D. (2018). Application of Machine Learning to Classification of Volcanic Deformation in Routinely Generated InSAR Data. *Journal of Geophysical Research: Solid Earth*, 123(8), 6592–6606. <https://doi.org/10.1029/2018JB015911>

Anantrasirichai, Nantheera, Biggs, J., Kelevitz, K., Sadeghi, Z., Wright, T., Thompson, J., Achim, A., & Bull, D. (2020). Deep learning framework for detecting ground deformation in the built environment using satellite insar data. *ArXiv*, 1–22.

Attema, E. (2005). Mission Requirements Document for the European Radar Observatory Sentinel-1. *Elements*, 31, 1–31.
https://esamultimedia.esa.int/docs/GMES/GMES_SENT1_MRD_1-4_approved_version.pdf
http://esamultimedia.esa.int/docs/GMES/GMES_SENT1_MRD_1-4_approved_version.pdf

Ayodele, T. O. (2010). Types of Machine Learning Algorithms. In *Intech*.
<https://doi.org/10.5772/9385>

- Balz, T., & Haala, N. (2002). Sar-Based 3D-Reconstruction of Complex Urban Environments. *Environments*.
- Biggs, J., & Wright, T. J. (2020). How satellite InSAR has grown from opportunistic science to routine monitoring over the last decade. *Nature Communications*, 11(1), 10–13. <https://doi.org/10.1038/s41467-020-17587-6>
- Brengman, C. M. J., & Barnhart, W. D. (2021). Identification of Surface Deformation in InSAR Using Machine Learning. *Geochemistry, Geophysics, Geosystems*, 22(3), e2020GC009204. <https://doi.org/https://doi.org/10.1029/2020GC009204>
- Chaussard, E., Wdowinski, S., Cabral-Cano, E., & Amelung, F. (2014). Land subsidence in central Mexico detected by ALOS InSAR time-series. *Remote Sensing of Environment*, 140, 94–106. <https://doi.org/10.1016/j.rse.2013.08.038>
- Chen, Y., Zhang, G., Ding, X., & Li, Z. (2000). Monitoring earth surface deformations with InSAR technology: principles and some critical issues. *Journal of Geospatial Engineering*, 2(1), 3–22.
- Choi, D., Shallue, C. J., Nado, Z., Lee, J., Maddison, C. J., & Dahl, G. E. (2019). On empirical comparisons of optimizers for deep learning. *ArXiv*, 1.
- Dong, L., & Shan, J. (2013). A comprehensive review of earthquake-induced building damage detection with remote sensing techniques. *ISPRS Journal of Photogrammetry and Remote Sensing*, 84, 85–99. <https://doi.org/10.1016/j.isprsjprs.2013.06.011>
- Fidon, L., Li, W., Garcia-Peraza-Herrera, L. C., Ekanayake, J., Kitchen, N., Ourselin, S., & Vercauteren, T. (2018). Generalised wasserstein dice score for imbalanced multi-class segmentation using holistic convolutional networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10670 LNCS, 64–76. https://doi.org/10.1007/978-3-319-75238-9_6
- François Chollet. (2018). Deep Learning with Python. In *The Lancet Neurology* (Vol. 10, Issue 2). [https://doi.org/10.1016/s1474-4422\(11\)70007-2](https://doi.org/10.1016/s1474-4422(11)70007-2)
- Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., & Lu, H. (2018). Dual attention network for scene segmentation. *ArXiv*, 3146–3154.
- Gombert, B., Duputel, Z., Jolivet, R., Doubre, C., Rivera, L., & Simons, M. (2018). Revisiting the 1992 Landers earthquake: A Bayesian exploration of co-seismic slip and off-fault damage. *Geophysical Journal International*, 212(2), 839–852.

<https://doi.org/10.1093/gji/ggx455>

- Guo, H., Liu, H., Wu, C., Zhi, W., Xiao, Y., & She, W. (2016). Logistic discrimination based on G-mean and F-measure for imbalanced problem. *Journal of Intelligent and Fuzzy Systems*, 31(3), 1155–1166. <https://doi.org/10.3233/IFS-162150>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016a). Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem*, 770–778. <https://doi.org/10.1109/CVPR.2016.90>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016b). Identity mappings in deep residual networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9908 LNCS, 630–645. https://doi.org/10.1007/978-3-319-46493-0_38
- Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). Communicated by Yann Le Cun A Fast Learning Algorithm for Deep Belief Nets 500 units 500 units. *Neural Computation*, 18(July), 1527–1554. <https://doi.org/10.1162/neco.2006.18.7.1527>
- Hoo, Z. H., Candlish, J., & Teare, M. D. (2017). What is a ROC curve? *Emergency Medicine Journal*.
- Hou, L., & Shi, P. (2011). Haiti 2010 earthquake—How to explain such huge losses? *International Journal of Disaster Risk Science*, 2(1), 25–33. <https://doi.org/10.1007/s13753-011-0003-x>
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, 1, 448–456.
- Janocha, K., & Czarnecki, W. M. (2016). On loss functions for deep neural networks in classification. *Schedae Informaticae*, 25, 49–59. <https://doi.org/10.4467/20838476SI.16.004.6185>
- Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–15.
- Kirscht, M., & Rinke, C. (1998). 3D reconstruction of buildings and vegetation from SAR images. *Proceedings of IAPR Workshop on Machine Vision Applications*, 228–231.

- Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *ACM International Conference Proceeding Series*, 145–151. <https://doi.org/10.1145/3383972.3383975>
- Lang, M. W., & Kasischke, E. S. (2008). Using C-band synthetic aperture radar data to monitor forested wetland hydrology in Maryland's coastal plain, USA. *IEEE Transactions on Geoscience and Remote Sensing*, 46(2), 535–546. <https://doi.org/10.1109/TGRS.2007.909950>
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>
- Lee, C. Y., & Lee, Z. J. (2012). A novel algorithm applied to classify unbalanced data. *Applied Soft Computing Journal*, 12(8), 2481–2485. <https://doi.org/10.1016/j.asoc.2012.03.051>
- Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. *ArXiv*, 2980–2988.
- LIU, C. an, CHEN, Z. xin, SHAO, Y., CHEN, J. song, Hasi, T., & PAN, H. zhu. (2019). Research advances of SAR remote sensing for agriculture applications: A review. *Journal of Integrative Agriculture*, 18(3), 506–525. [https://doi.org/10.1016/S2095-3119\(18\)62016-7](https://doi.org/10.1016/S2095-3119(18)62016-7)
- Ma, P., Zhang, F., & Lin, H. (2019). Prediction of InSAR time-series deformation using deep convolutional neural networks. *Remote Sensing Letters*, 11(2), 137–145. <https://doi.org/10.1080/2150704X.2019.1692390>
- Marino, A., Dierking, W., & Wesche, C. (2016). A Depolarization Ratio Anomaly Detector to Identify Icebergs in Sea Ice Using Dual-Polarization SAR Images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(9), 5602–5615. <https://doi.org/10.1109/TGRS.2016.2569450>
- Mikołajczyk, A., & Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. *2018 International Interdisciplinary PhD Workshop, IIPhDW 2018*, 117–122. <https://doi.org/10.1109/IIPHDW.2018.8388338>
- Moro, M., Saroli, M., Stramondo, S., Bignami, C., Albano, M., Falcucci, E., Gori, S., Doglioni, C., Polcari, M., Tallini, M., Macerola, L., Novali, F., Costantini, M., Malvarosa, F., & Wegmüller, U. (2017). New insights into earthquake precursors from InSAR. *Scientific Reports*, 7(1), 1–11. <https://doi.org/10.1038/s41598-017->

12058-3

- N.Miranda, G.Hajduch, & N.Franceschi. (2015). *Sentinel-1 Level-0 Data Decoding Package*. 33(0), 1–15.
- Narayan, S. (1997). The Generalized Sigmoid Activation Function: Competitive Supervised Learning. *Information Sciences*, 99(1–2), 69–82.
[https://doi.org/10.1016/S0020-0255\(96\)00200-9](https://doi.org/10.1016/S0020-0255(96)00200-9)
- Ng, A. Y. (2004). L1 and L2 regularisation comparison. *Proceedings of the 21 St International Conference on Machine Learning*.
- Nicolas, J. M., & Adragna, F. (2000). The Principles of Synthetic Aperture Radar. *Processing of Synthetic Aperture Radar Images*, 25–55.
<https://doi.org/10.1002/9780470611111.ch2>
- Nowozin, S. (2014). Optimal decisions from probabilistic models: The intersection-over-union case. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 548–555. <https://doi.org/10.1109/CVPR.2014.77>
- Ofeigsson, B. G., Hooper, A., Sigmundsson, F., Sturkell, E., & Grapenthin, R. (2011). Deep magma storage at Hekla volcano, Iceland, revealed by InSAR time series analysis. *Journal of Geophysical Research: Solid Earth*, 116(5), 1–15.
<https://doi.org/10.1029/2010JB007576>
- Padalia, H., & Yadav, S. (2017). Evaluation of RISAT-1 SAR data for tropical forestry applications. *Advances in Space Research*, 59(1), 2–11.
<https://doi.org/10.1016/j.asr.2016.08.026>
- Pavelka, K., Šedina, J., Matoušková, E., Hlaváčová, I., & Korth, W. (2019). Examples of different techniques for glaciers motion monitoring using InSAR and RPAS. *European Journal of Remote Sensing*, 52(sup1), 219–232.
<https://doi.org/10.1080/22797254.2018.1559001>
- Potin, P., Rosich, B., Miranda, N., Grimont, P., Shurmer, I., O'Connell, A., Krassenburg, M., & Gratadour, J. B. (2019). Copernicus Sentinel-1 Constellation Mission Operations Status. *International Geoscience and Remote Sensing Symposium (IGARSS), April 2016*, 5385–5388.
<https://doi.org/10.1109/IGARSS.2019.8898949>
- Raab, S. (2016). *Development and Implementation of an Efficient Temperature Management System for SAR System Calibration Transponders*.

- Rouet-Leduc, B., Jolivet, R., Dalaison, M., Johnson, P. A., & Hulbert, C. (2020). *Autonomous Extraction of Millimeter-scale Deformation in InSAR Time Series Using Deep Learning*. <http://arxiv.org/abs/2012.13849>
- Schlögel, R., Doubre, C., Malet, J. P., & Masson, F. (2015). Landslide deformation monitoring with ALOS/PALSAR imagery: A D-InSAR geomorphological interpretation method. *Geomorphology*, 231, 314–330. <https://doi.org/10.1016/j.geomorph.2014.11.031>
- Schneider, F., Balles, L., & Hennig, P. (2019). Deepobs: A deep learning optimizer benchmark suite. *7th International Conference on Learning Representations, ICLR 2019*, 1–14.
- Sharma, S., Sharma, S., & Anidhya, A. (2020). Understanding Activation Functions in Neural Networks. *International Journal of Engineering Applied Sciences and Technology*, 4(12), 310–316.
- Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.
- Sorrentino, L., Cattari, S., da Porto, F., Magenes, G., & Penna, A. (2019). Seismic behaviour of ordinary masonry buildings during the 2016 central Italy earthquakes. *Bulletin of Earthquake Engineering*, 17(10), 5583–5607. <https://doi.org/10.1007/s10518-018-0370-4>
- Srivastava, N., Hinton, G., Krizhevsky, A., & Sutskever, I. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfittin. *Journal of Machine Learning Research*, 15(3–4), 1930–1958. [https://doi.org/10.1016/0370-2693\(93\)90272-J](https://doi.org/10.1016/0370-2693(93)90272-J)
- Sun, J., Wauthier, C., Stephens, K., Gervais, M., Cervone, G., La Femina, P., & Higgins, M. (2020). Automatic Detection of Volcanic Surface Deformation Using Deep Learning. *Journal of Geophysical Research: Solid Earth*, 125(9), 1–17. <https://doi.org/10.1029/2020JB019840>
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07-12-June*, 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the

- Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem*, 2818–2826. <https://doi.org/10.1109/CVPR.2016.308>
- Tello, M., López-Martínez, C., & Mallorqui, J. J. (2005). A novel algorithm for ship detection in SAR imagery based on the wavelet transform. *IEEE Geoscience and Remote Sensing Letters*, 2(2), 201–205. <https://doi.org/10.1109/LGRS.2005.845033>
- Valade, S., Ley, A., Massimetti, F., D'Hondt, O., Laiolo, M., Coppola, D., Loibl, D., Hellwich, O., & Walter, T. R. (2019). Towards global volcano monitoring using multisensor sentinel missions and artificial intelligence: The MOUNTS monitoring system. *Remote Sensing*, 11(13), 1–31. <https://doi.org/10.3390/rs11131528>
- Valueva, M. V., Nagornov, N. N., Lyakhov, P. A., Valuev, G. V., & Chervyakov, N. I. (2020). Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. *Mathematics and Computers in Simulation*, 177, 232–243. <https://doi.org/10.1016/j.matcom.2020.04.031>
- Van Hieu, N., & Hien, N. L. H. (2020). Recognition of plant species using deep convolutional feature extraction. *International Journal on Emerging Technologies*, 11(3), 904–910.
- Vincent, P., Bourbigot, M., Johnsen, H., & Piantanida, R. (2020). *Sentinel-1 Product Specification*.
- Wang, Y., Guo, Y., Hu, S., Li, Y., Wang, J., Liu, X., & Wang, L. (2019). Ground deformation analysis using InSAR and backpropagation prediction with influencing factors in Erhai Region, China. *Sustainability (Switzerland)*, 11(10). <https://doi.org/10.3390/su11102853>
- Weng, W., & Zhu, X. (2015). UNet: Convolutional Networks for Biomedical Image Segmentation. *IEEE Access*, 9, 16591–16603. <https://doi.org/10.1109/ACCESS.2021.3053408>
- Wilson, A. C., Roelofs, R., Stern, M., Srebro, N., & Recht, B. (2017). The marginal value of adaptive gradient methods in machine learning. *Advances in Neural Information Processing Systems, 2017-Decem(Nips)*, 4149–4159.

Attachments

1. E-poster for Dragon Symposium 2021: “Unbalanced Technics to Improve the Train for ML Models to Detect Earthquake Fringes”; Bruno Silva, Joaquim Sousa, Milan Lazecky, António Cunha.
2. E-poster for Dragon Symposium 2021: “ML Segmentation Models to Automatically Identify Areas Affected by Earthquakes”; Bruno Silva, Joaquim Sousa, Milan Lazecky, António Cunha.

Unbalanced Technics to Improve the Train for ML Models to Detect Earthquake Fringes

Bruno Silva | Joaquim Sousa | Milan Lazecky | António Cunha

Abstract

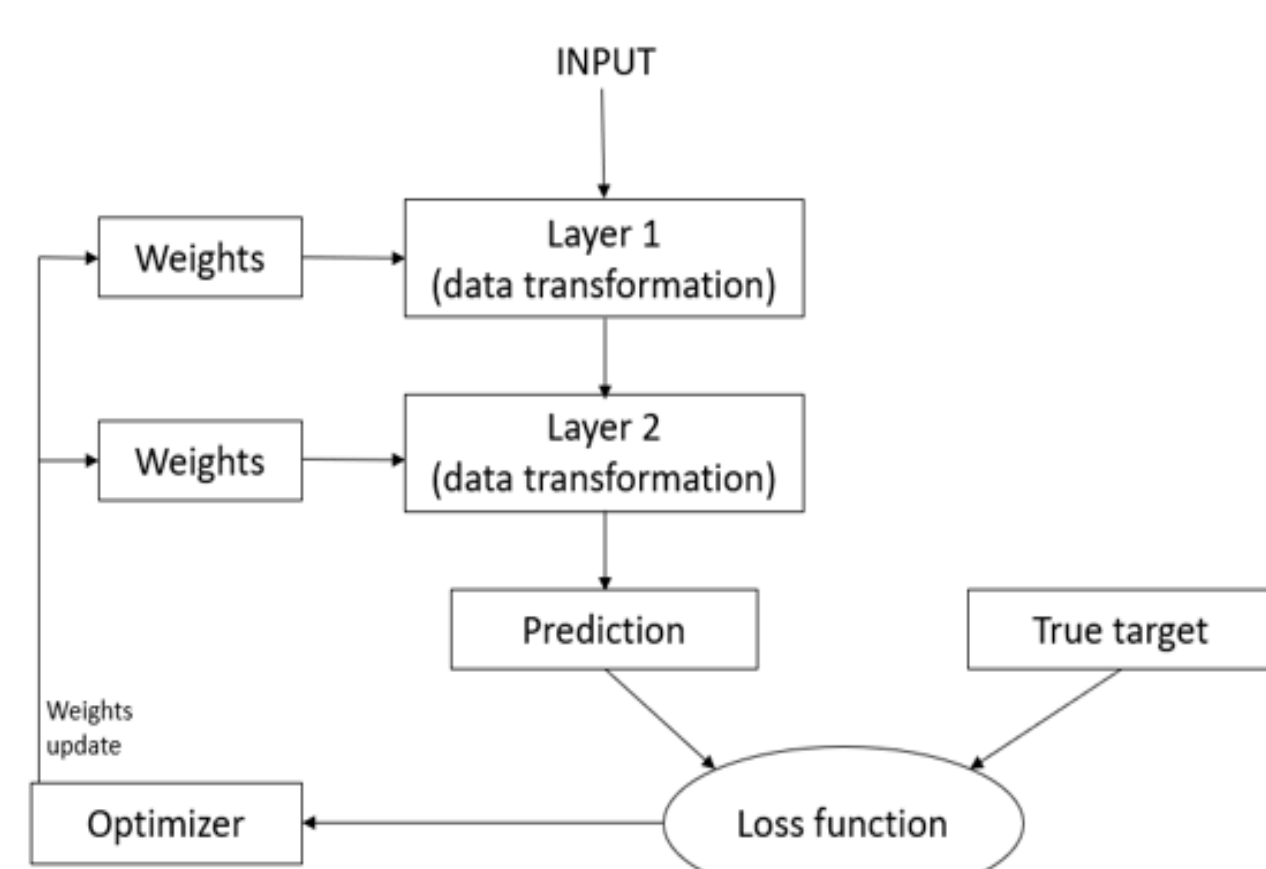
Machine Learning can automatically process large datasets in the most varied areas, including remote sensing data, and it has become an opportunity for earth observation. Recent studies have demonstrated the ability to detect visible fringes deformations in InSAR images. However, InSAR data is frequently unbalanced - deformations are sparse compared to those that do not have deformation, and it needs special attention for training ML models.

In this work, we created two InSAR datasets with 29 earthquake cases from the LICS database. At start we use Data Augmentation to deal with data unbalanced to detect fringes, but when the data grew, and the unbalancing got bigger DA start to perform worse, so we apply a new technique to deal with the unbalancing.

Introduction

Train deep learning models

1. prepare the input data
2. The data will pass for the layers, resulting a prediction
3. Prediction are evaluated through loss function resultin a score.
4. The score is used as feedback to adjust model weights through the optimizer.

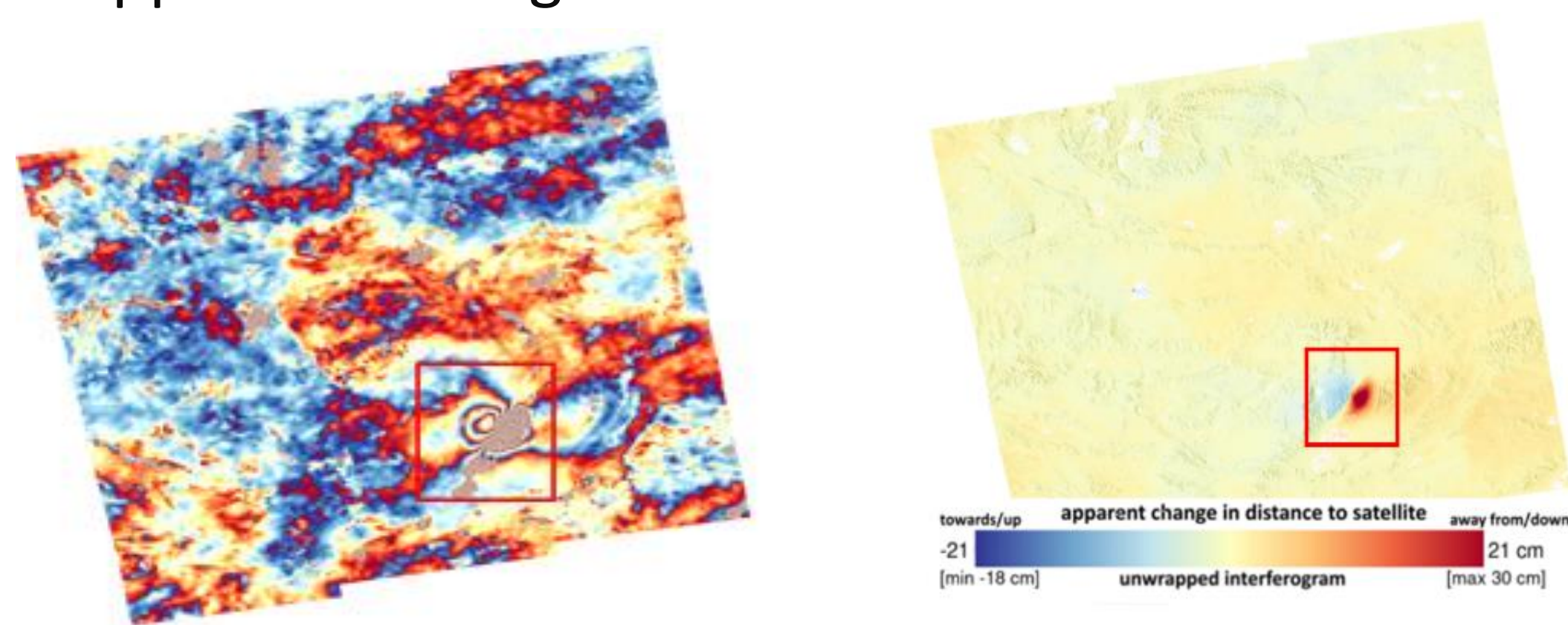


Focal Loss compensate data less represented.

Data augmentation creates artificial data with small alteration, to balance the data in the same amount.

Objective

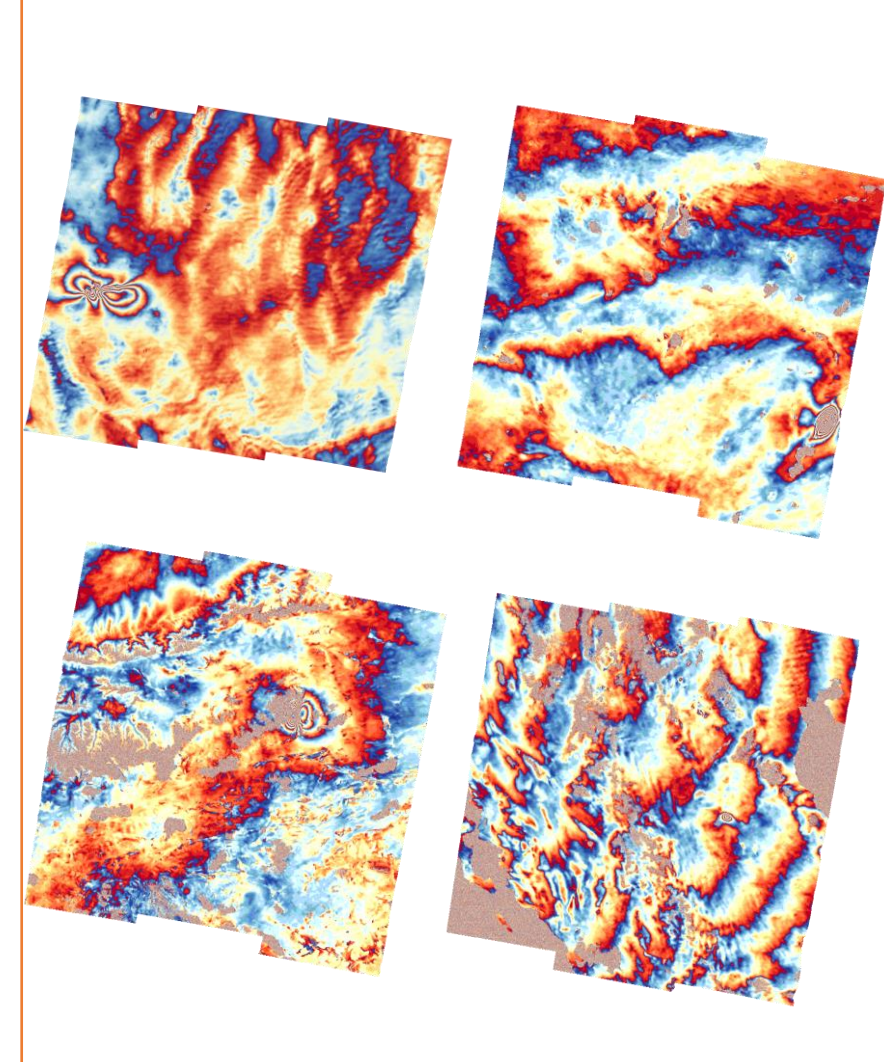
Deal with unbalanced data training deep learning models to identify deformation in InSAR images, both is wrapped and unwrapped interferograms.



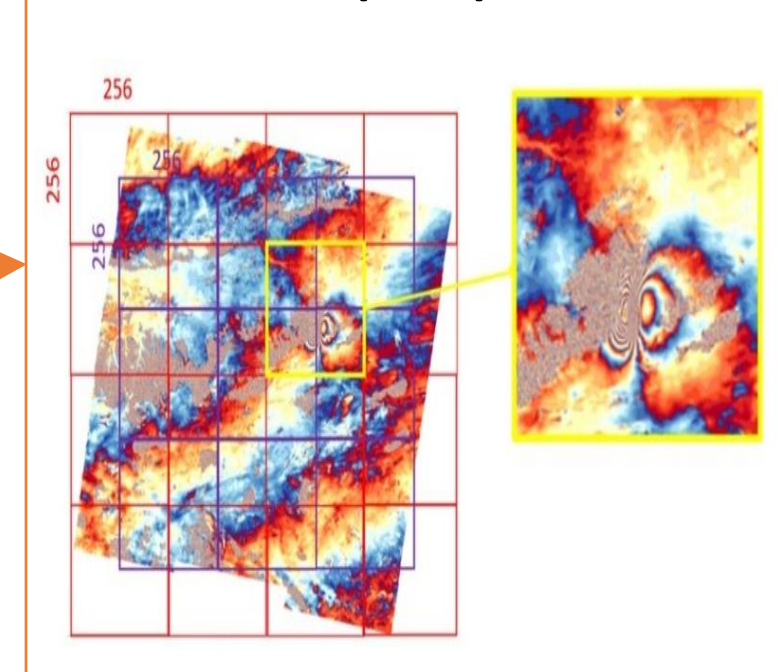
Methods

Two dataset of InSAR interferograms were created (wrapped and unwrapped). We cut images into 256x256pixels overlapped patches. Finally we use the patches to train 3 pre-trained models with focal loss and we use the best model to compare focal loss with data augmentation.

Dataset Creation



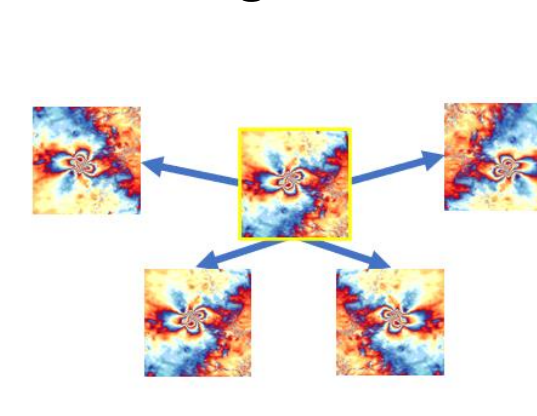
Dataset preparation



Classification with Focal Loss

$$FL(pt) = -\alpha_c(1-pt)^{\gamma}\log(pt)$$

Classification with Data Augmentation

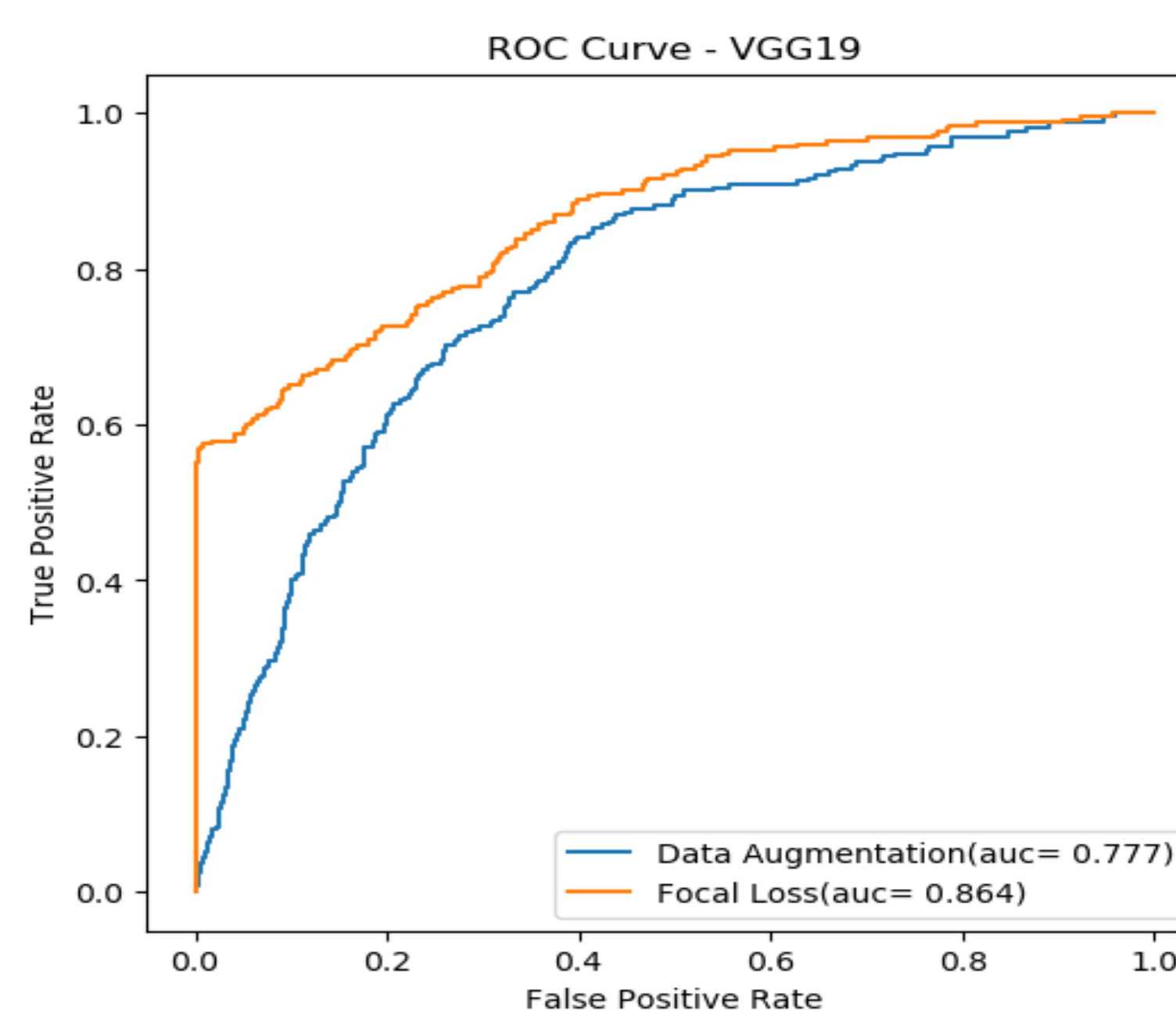
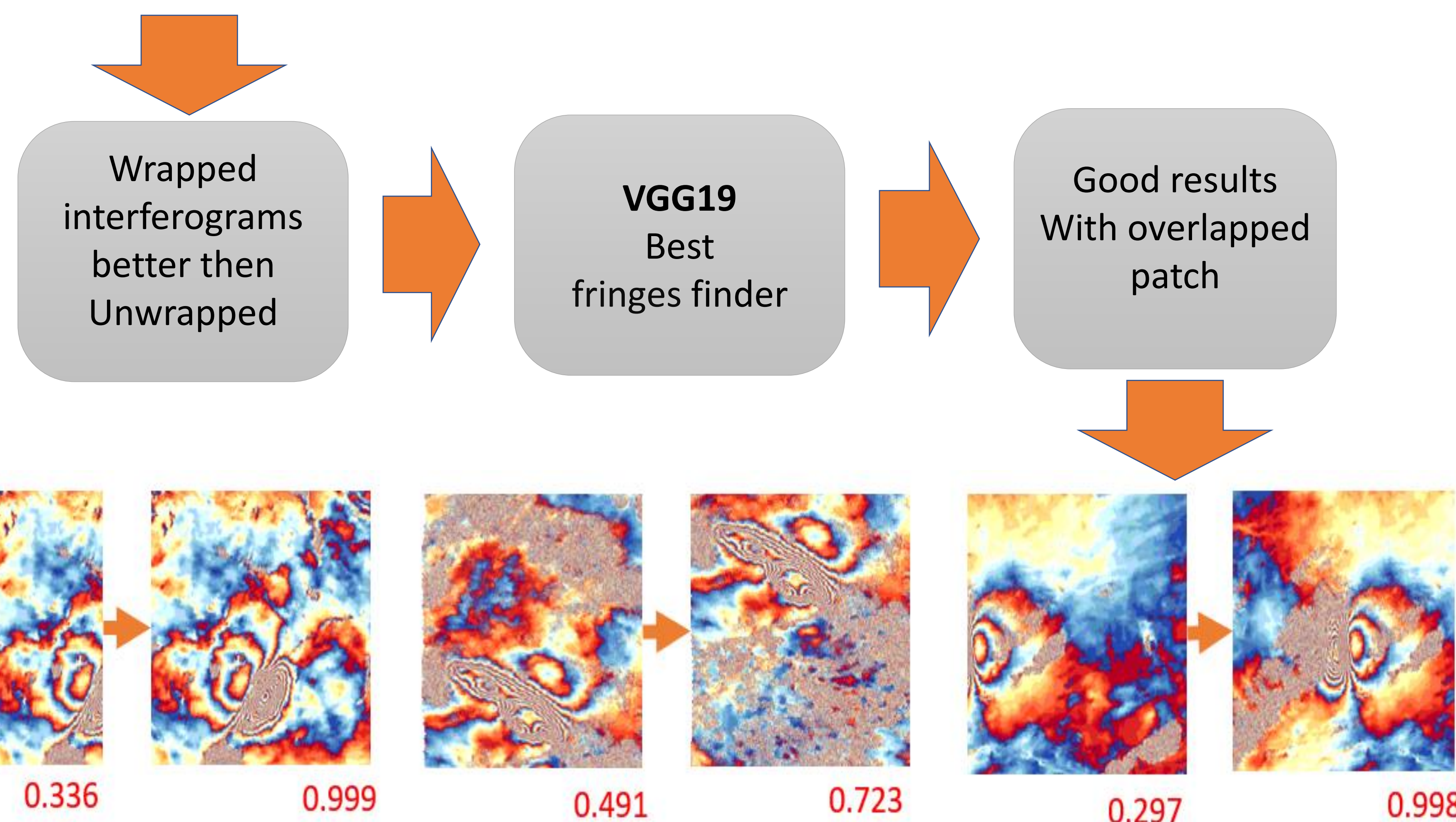
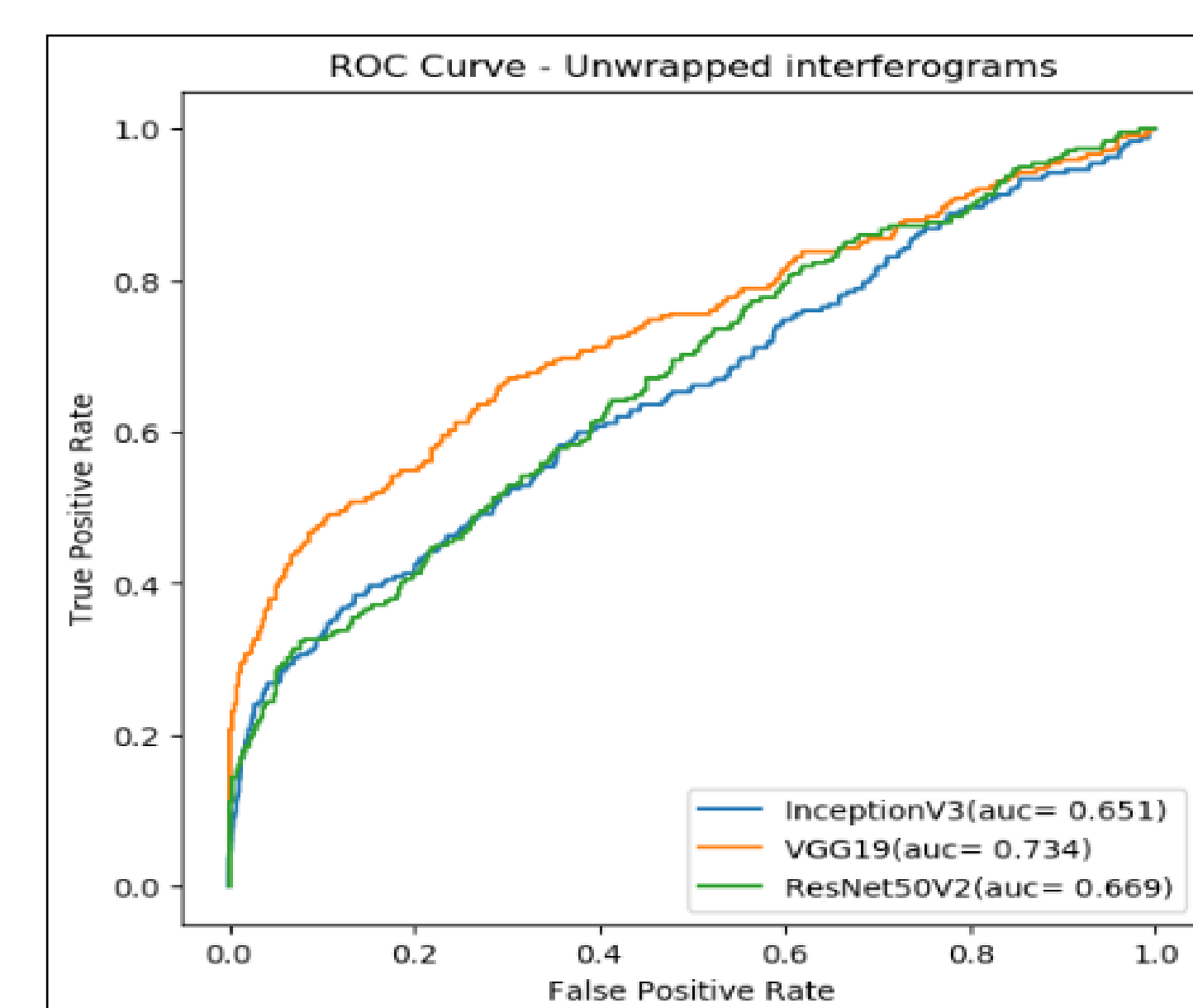
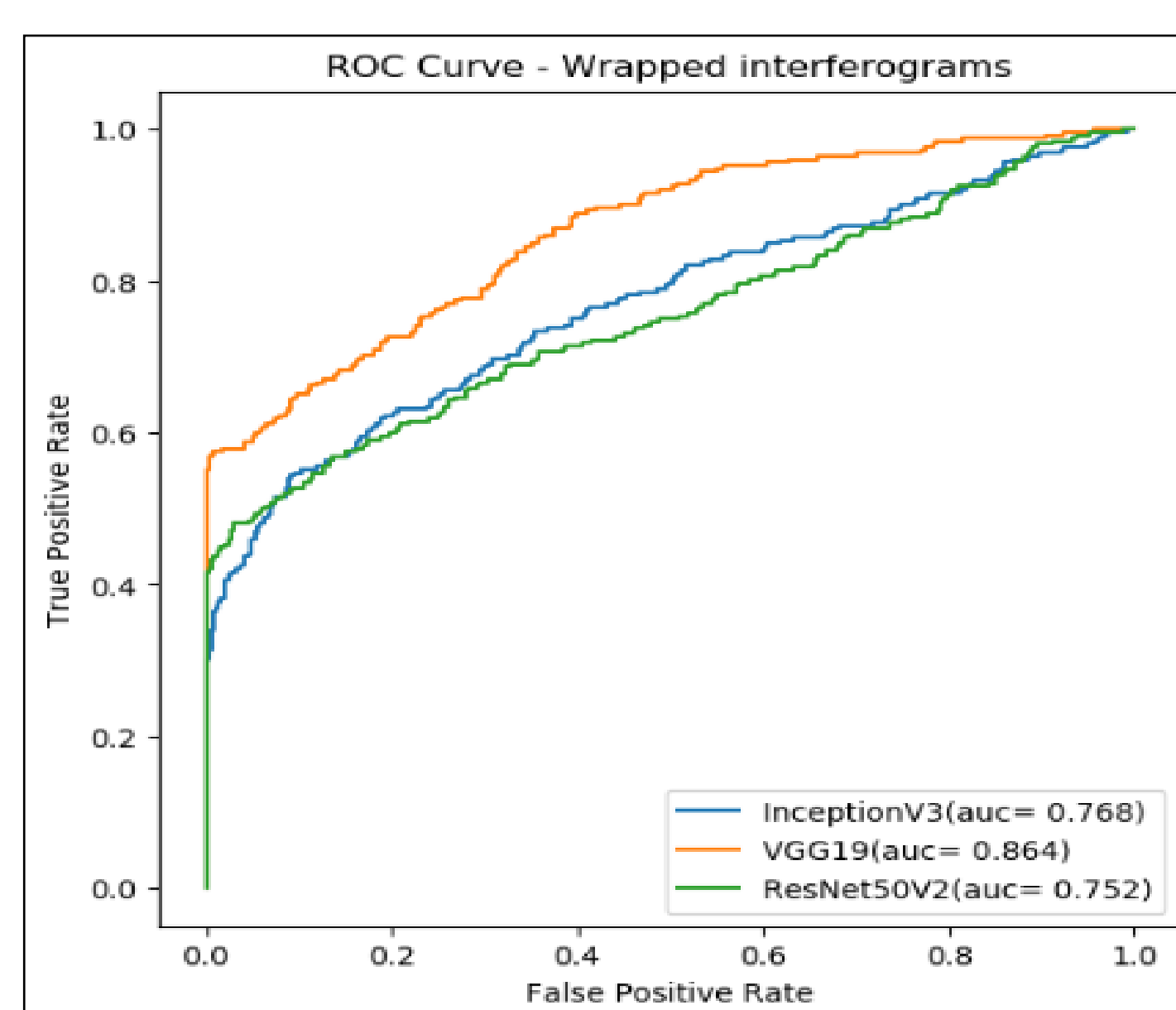


Dataset

	Train	Validation	Test
Earthquake fringes (deformation)	499	380	252
No deformation	14979	4051	3826

Models evaluation

Models	Wrapped interferograms			Unwrapped interferograms		
	Accuracy	F1 Score	AUC	Accuracy	F1 Score	AUC
InceptionV3	0.952	0.371	0.768	0.938	0.137	0.651
VGG19	0.971	0.691	0.864	0.951	0.330	0.734
Resnet50V2	0.960	0.526	0.752	0.943	0.160	0.669



Focal loss performs better than data augmentation to deal with unbalanced data

Conclusion

1. We successful create two InSAR datasets.
2. Wrapped interferograms proves to be better to train deep learning models.
3. VGG19 was the best model to detect earthquake deformation fringes
4. Focal Loss proves to be better to deal with data unbalanced then data augmentation.
5. We consider that a bigger dataset with more earthquake cases can improve these results.
6. Knowing the reasons why some patches have bad classification comparing with the overlapped ones we can create a well chosen patches to train and improve the results.

ML Segmentation Models to Automatically Identify Areas Affected by Earthquakes

Bruno Silva | Joaquim Sousa | Milan Lazecky | António Cunha

Abstract

Some studies have been applying ML's ability to detect InSAR images with the fringes in InSAR images; however, no studies have been found where the area is isolated using segmentation techniques. In this work we apply segmentation models to identify areas affected by earthquakes in InSAR interferograms.

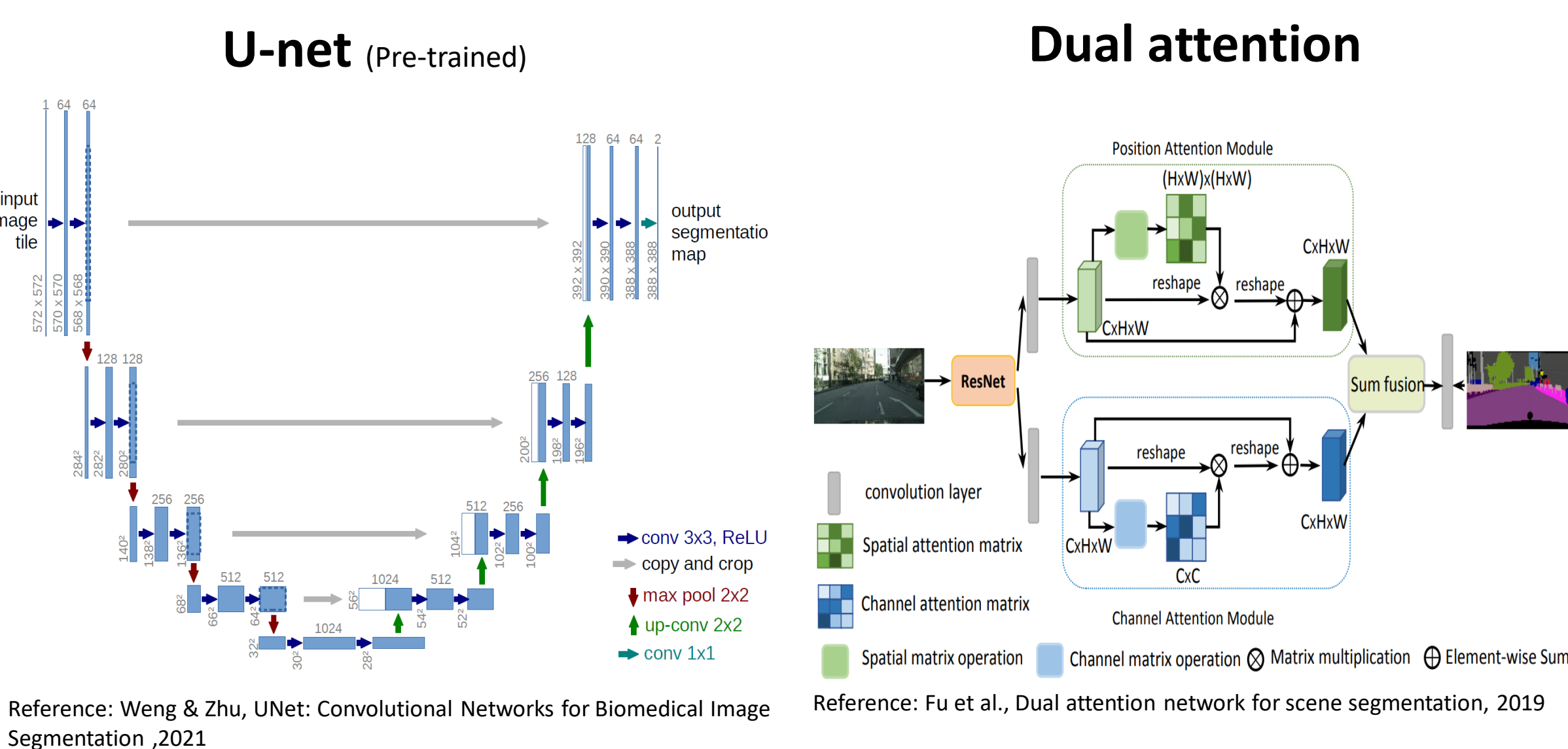
Introduction

Train deep learning models

1. prepare the input data
2. The data will pass for the layers, resulting a prediction
3. Prediction are evaluated through loss function resulting a score.
4. The score is used as feedback to adjust model weights throught the optimizer.

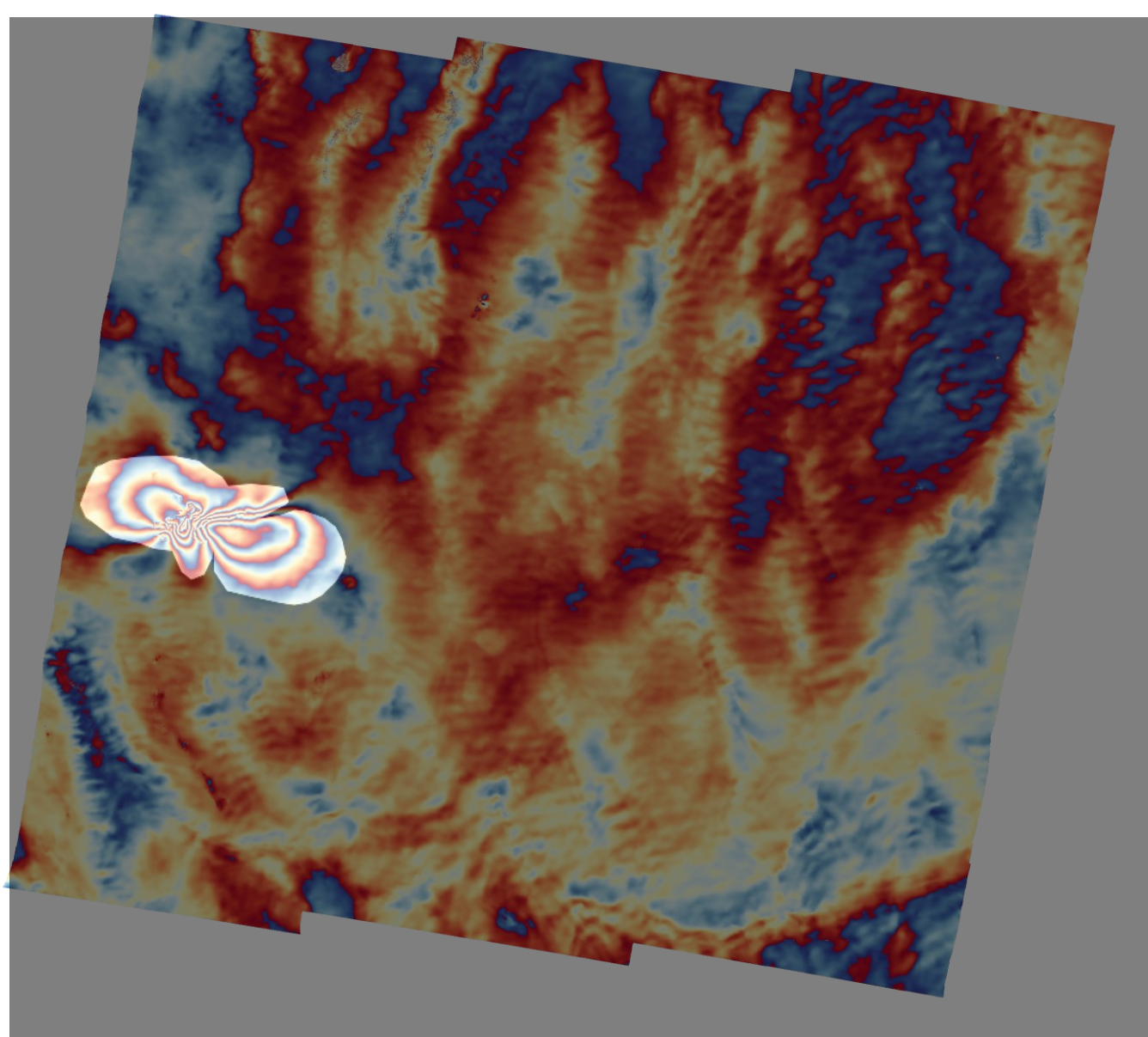
U-net uses an encoder and a deccoder to segment some área.

Dual attention is a technique where integrate local features with their global dependencies based on the self-attention mechanism.



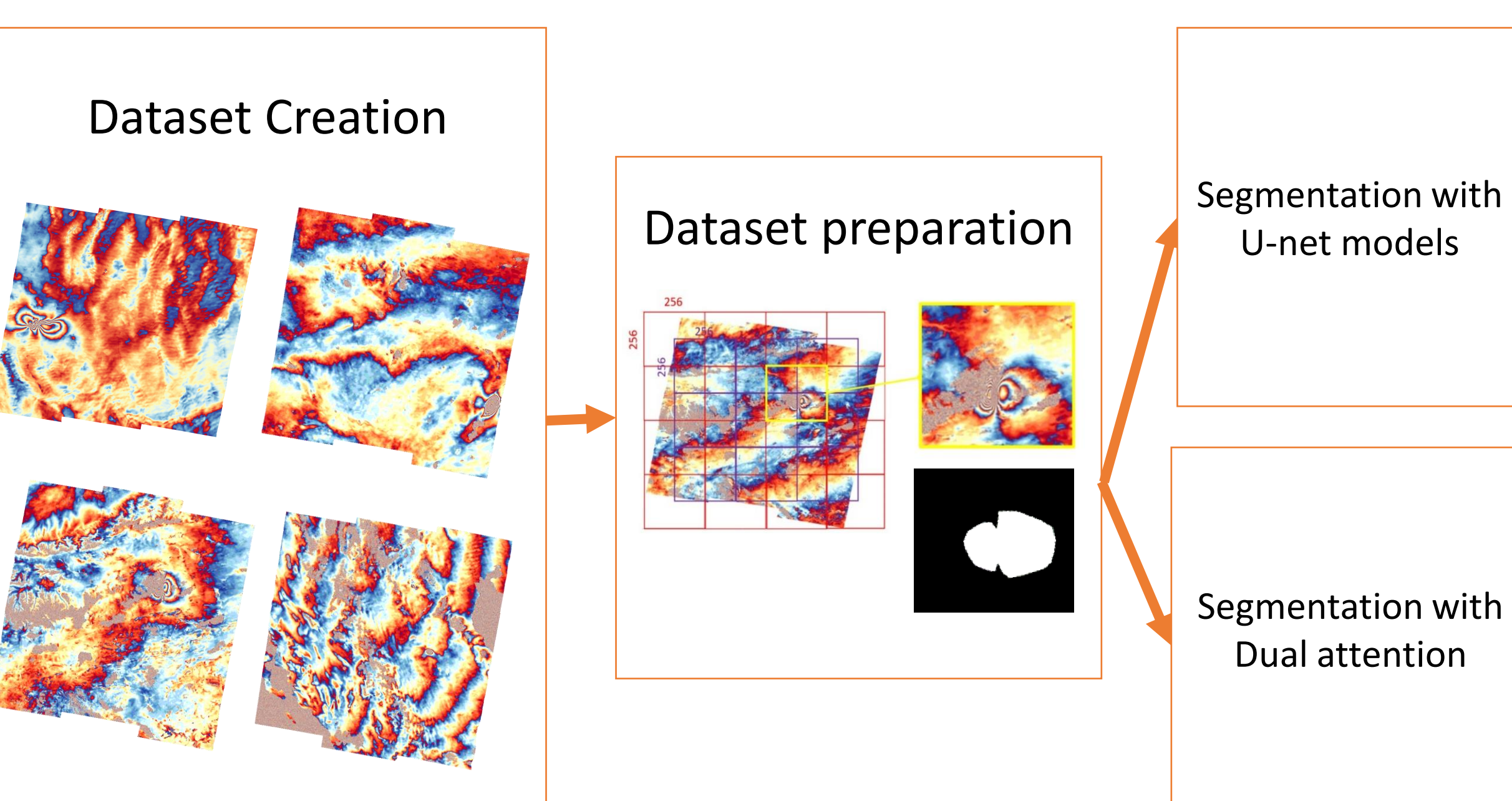
Objective

Evaluation Deep learning segmentation techniques to estimate the area affected by earthquakes in wrapped interferograms without atmospheric correction.



Methods

We start this work creating a InSAR dataset with 469 interferograms from 29 earthquake cases between 2019 and 2021 and create a mask for each of them. We create the mask with a margin to test if the model can detect deformation that cant be seen due to the atmospheric error. we then cut the images into 256x256pixels overlapped and choose the ones with fringe. Finally we use the patches to train 3 pre-trained U-net models and a dual attention model.



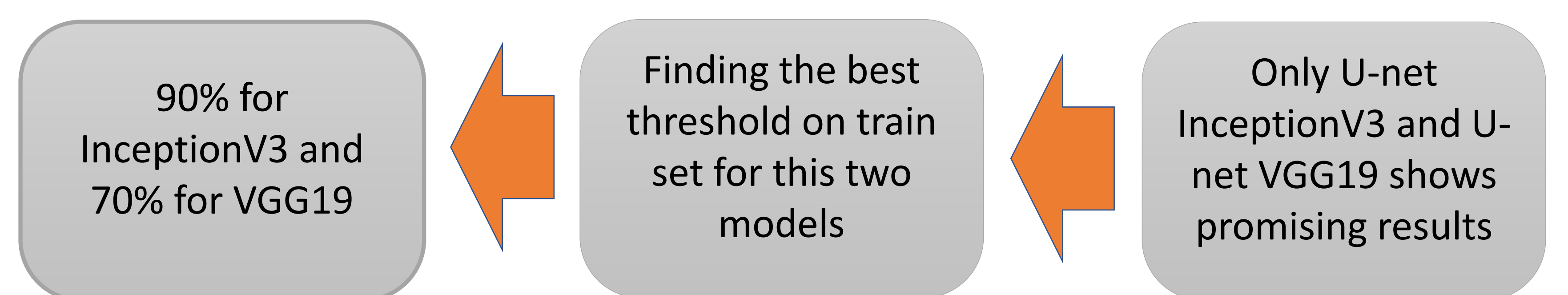
Dataset

	Train	Validation	Test
Earthquake fringes (deformation)	499	380	252

Models evaluation

First test with 50% threshold

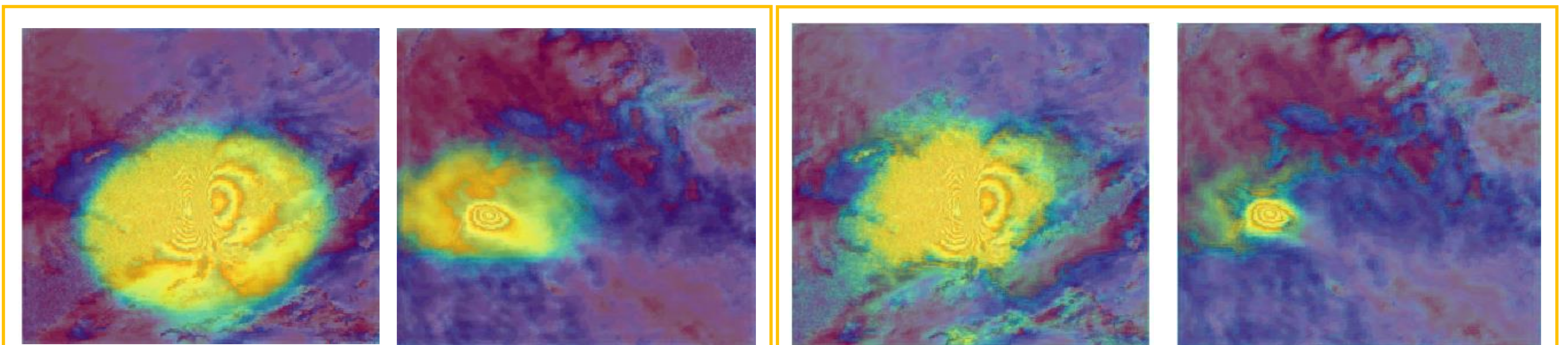
	Model	IoU Score	Dice Score	Accuracy
U-net	InceptionV3	0.43	0.59	0.83
	VGG19	0.32	0.47	0.85
	Resnet50	0.13	0.22	0.80
Dual attention	Resnet50	0.25	0.39	0.84



U-Net Models	IoU Score	Dice Score	Accuracy
Coder: InceptionV3	0.48	0.63	0.87
Coder: VGG19	0.47	0.62	0.85

U-Net with InceptionV3 coder can found fringes but is not good to detect margins

U-Net with VGG19 coder better find fringes and margins but interpret noise as fringes



Predicted segmenation probability maps: left U-Net with InceptionV3 coder; right U-Net with VGG19 coder

Conclusion

1. We successful create two InSAR datasets with area affected by earthquakes annotated.
2. VGG19 and InceptionV3 for U-Net model were the best models to segment earthquake deformation fringes.
3. Deep learning proves to be able to "see" deformation fringes and locate them.
4. Dual attention had worse results then U-net, not being good for segment deformation fringes in InSAR interferograms.

Future work

- Results were promising but bigger dataset is needed to improve results.
- New segmentation models or combination of models may improve results..