SENECA: Change Detection in Optical Imagery using Siamese Networks with Active-Transfer Learning

Giuseppina Andresini[a,b,*], Annalisa Appice[a,b], Dino Ienco[c], Donato Malerba[a,b]

[a] *Department of Computer Science, University of Bari Aldo Moro, Bari, Italy*
[b] *Consorzio Interuniversitario Nazionale per l'Informatica - CINI, Bari, Italy*
[c] *INRAE, UMR TETIS, University of Montpellier, Montpellier, France*

Abstract

[1] Change Detection (CD) aims to distinguish surface changes based on bi-temporal remote sensing images. In the recent years, deep neural models have made a breakthrough in CD processes. However, training a deep neural model requires a large volume of labelled training samples that are time-consuming and labour-intensive to acquire. With the aim of learning an accurate CD model with limited labelled data, we propose SENECA: a method based on a CD Siamese network, which takes advantage of both Active Learning (AL) and Transfer Learning (TL) to handle the constraint of limited supervision. More precisely, we jointly use AL and TL to adapt a CD model trained on a labelled source domain to a (related) target domain featured by a restricted access to labelled data. We report results from an experimental evaluation involving five pairs of images acquired via Sentinel-2 satellites between 2015 and 2018 in various locations picked all over Asia and USA. The results show the beneficial effects of the proposed AL and TL strategies on

* Corresponding author

Email addresses: `giuseppina.andresini@uniba.it` (Giuseppina Andresini), `annalisa.appice@uniba.it` (Annalisa Appice), `dino.ienco@inrae.fr` (Dino Ienco), `donato.malerba@uniba.it` (Donato Malerba)

the accuracy of the decisions made by the CD Siamese network and depict the merit of the proposed approach over competing CD baselines.

Keywords: Active learning, Transfer learning, Siamese network, Change detection, Sentinel-2 data

1. Introduction

Change Detection (CD) is a central task in the field of computer vision since it has the objective to detect changes in multiple images of the same scene acquired at different period of time (Ru et al. 2021). Focusing on the analysis of optical remote sensing images depicting the same geographical area, the CD task is the process of detecting differences among various images of the same scene as a consequence of natural and/or human activities.

Due to the unprecedented availability of remote sensing imagery acquired by up-to-date Earth observation systems (a notable example is the Sentinel-2 mission belonging to the European Copernicus Programme [2]), it is becoming easier and easier to obtain images covering the same geographical area acquired with a regular revisit time. This technological revolution highlights the importance of conceiving and developing effective CD methods to fully exploit the amount of freely available remote sensing information.

Application-wise, CD methods are largely employed in remote sensing analysis (Lv et al. 2022) to cope with a diverse set of applications like land cover change detection (Shi, Zhong, Zhao, Lv, Liu & Zhang 2022), urban change detection (Hafner et al. 2022), disaster management (Sublime & Kalinicheva 2019) and environmental monitoring (Lewis et al. 2016), among the others.

Modern advances in CD methods mainly rely on deep learning (DL) approaches (Jiang et al. 2022) due to their ability to cope with imagery data through the automatic extraction of hierarchical multilevel features via representational learning (Bengio et al. 2013). Despite remarkable performances exhibited by neural network approaches in many different applications of image analysis, one of the main limitation of the use of DL methods is related to the label-hungry behaviour they exhibit. In fact, a large amount of labelled samples is, commonly, required for effective deep neural network

[2] https://www.copernicus.eu/en

training, while facing this condition poses critical issues related to the deployment of DL approaches in tasks characterised by limited amount of labelled data (Ouali et al. 2020). For the specific case of remote sensing CD, it can be highly labor-intensive and time-consuming to collect remote sensing image pairs with well-labelled change information each time a CD method must be deployed. This condition may affect the use of DL methods for CD, since a DL model may need to be transferred from the imagery data (pair of images) on which it is learnt (source data) to new unseen imagery data (target data). Moreover, the crucial point is how to alleviate the dependence of DL models from large amount of labelled data, while meeting the requirement to transfer CD models from a source to a target scenario.

With the objective to reduce the dependence of remote sensing CD models from the necessity to access abundant amount of labelled data when transferred on new scene, in this paper, we propose SENECA (Siamese nEtwork based chaNge detection in optical imagEry with aCtive trAnsfer learning): a method based on a Siamese network, which combines both Active Learning (AL) and Transfer Learning (TL) to effectively deal with remote sensing CD analysis in a scenario featured by limited supervision.

The Siamese network is a neural model especially tailored to compare together pairs of entities (Lu et al. 2017) with the aim to learn data embeddings that satisfy pair-wise metric constraints. As a Siamese network is well-suited to deal with the class imbalance condition (Gautheron et al. 2020), recent studies (Shi, Liu, Li, Liu, Wang & Zhang 2022, Ruzicka et al. 2020) have started the investigation of Siamese networks in CD tasks, where the number of changed pixels is often much less than that of unchanged ones.

The proposed method firstly learns a CD Siamese network on source data, where change labelled data are available and, successively, it adapts the source Siamese network to the target data through a Transfer Learning (TL) strategy. TL is performed with fine tuning, that is one of the most widely used approach for TL when working with DL models. In particular, the fine tuning approach starts with a pre-trained deep neural model on the source data and trains it further on the target data. In this study, the fine tuning approach is performed with a limited supervision provided by means of an Active Learning (AL) strategy. More precisely, we adopt a segmentation-based AL strategy that allows the Siamese network pre-trained for CD in a source area to select samples that spatially span the target area. This may contribute to reduce possible issues exhibited to confidence-based AL strategy in remote sensing data analysis (Pasolli et al. 2014) that are more prone to select redundant, in terms of spatial auto-correlation, samples.

The experimental evaluation, involving recent state of the art CD competitors, on five pair of images acquired via the Sentinel-2 satellite missions [3] between 2015 and 2018 in various locations picked all over Asia and USA has demonstrated the quality and the value of the proposed approach. More precisely, the results show the beneficial effects of combining AL and TL for all the downstream CD tasks.

In short, this paper provides the following contributions:

- The definition of a new CD method that is formulated combining both AL and TL, in order to reduce the necessity to access abundant amount of labelled samples when a CD neural network (Siamese network) is transferred from a source scene to a new target scene.
- The use of a segmentation-based AL strategy that allows us to effectively select active samples spanning the target area, in order to transfer the pre-trained CD Siamese network from the source to the target pair of images.
- An in-depth and extensive evaluation of the proposed method SENECA w.r.t. recent competing CD methods on five co-registered, bi-temporal multispectral images acquired with Sentinel-2 satellites in locations picked all over both Asia and USA.

The rest of this manuscript is organised as follows. Section 2 presents the recent related literature in remote sensing CD analysis. Section 3 introduces the background and the CD problem definition we adopt in our work. Section 4 describes the proposed Active-Transfer Learning (ATL) CD framework. Section 5 introduces the experimental settings, the performances evaluation as well as the discussion related to the results while Section 6 concludes and pave the way to possible future works.

2. Related Work

A general overview of CD approaches for land cover dynamics is presented in (Lv et al. 2022). Here, the authors review the main issues in terms of methods, applications and available benchmarks related to remote sensing

[3] https://sentinel.esa.int/nl/web/sentinel/missions/sentinel-2

CD with a particular focus on Very High spatial Resolution (VHR) imagery. Recently, Jiang et al. (2022) have provided a review of CD methods for remote sensing imagery under the lens of DL-based techniques underlying the fact that the community still lacks of a comprehensive review of the recent progress concerning neural network methods in remote sensing CD.

With a focus on the advances on unsupervised CD methods, Celik (2009) defined a PCA-based method for CD in multitemporal satellite images. This method partitions a difference image into non-overlapping blocks and performs the PCA, in order to extract the orthonormal eigenvectors of the set of non-overlapping blocks and build an eigenvector space. Subsequently, it represents each pixel of the difference image with a new feature vector that is the projection of the block-based difference image samples onto the generated eigenvector space. Finally, the CD map is built by partitioning the feature vector space into two clusters using k-means clustering with $k = 2$ and then assigning each pixel to the one of the two clusters by using the minimum Euclidean distance between the pixel's feature vector and mean feature vector of clusters.

Appice et al. (2020) introduced an unsupervised learning method for CD. This method combines clustering, PCA and classification with the aim to separate changed areas from unchanged background. More in detail, firstly a clustering stage is performed on the bi-temporal images with the aim to identify an initial set of labelled samples and, successively, the extracted labelled samples are used to feed a supervised binary classification stage. The classification stage trains a Random Forest from the principal components of the fusion (concatenation) of bi-temporal pixel vectors using the labels produced in the clustering stage.

López-Fandiño et al. (2019) experimented a change vector analysis (CVA) method in the field of imagery CD. The proposed CVA method computes the difference between two optical images of a scene with the spectral angle distance and uses the Otsu's thresholding to separate the changed areas from the unchanged areas. Andresini et al. (2022) investigated the use of autoencoder neural networks for CVA in a pair of optical images. More precisely, given a pair of optical images, the method learns an autoencoder model on the first image of the bi-temporal image pair then, the model is employed to reconstruct both the first and second images. Successively, the spectral angle distance is computed pixel-wise and, finally a threshold approach is adopted to separate changed from non-changed pixels in a totally unsupervised way.

Ma et al. (2019) illustrated a matrix factorisation method for CD in synthetic aperture radar images. In this method, the factorisation model of the low-rank and sparse matrix is used to extract both (unchanged) background and (changed) foreground information from images. More in detail, mean and variance matrices related to both unchanged and changed areas are summarised through statistical features that are, subsequently, used to learn a naive Bayes classifier. At the end, the classification model is employed to derive a CD map that distinguishes between changed and unchanged areas.

Wu et al. (2020) described an unsupervised method for CD in optical images based on generative adversarial networks. The proposed method uses CVA to build an initial CD map. Subsequently it applies a training sample selection method to select training samples that are processed to train the generative adversarial network. The generator of the generative adversarial network is used to build the final CD map.

Wu et al. (2021) illustrated an unsupervised method defined for CD in heterogeneous images. This method takes a pair of images acquired with different sensors (e.g., optical images and synthetic aperture radar images) as input. It combines together convolutional autoencoder and commonality autoencoder with the aim to firstly extract a vector-based representation of the input images and, successively, extract the common features by means of a reconstruction process. Finally, it deploys an unsupervised segmentation approach on the difference map to extract the changed areas.

Regarding recent supervised CD methods, Daudt et al. (2018) proposed a CD framework based on a fully convolutional Siamese network. In the proposed method, firstly the image pairs are encoded via a Siamese network with the aim to extract new data representation from each of the images and, then, the extracted bi-temporal representations are combined with the aim to produce a CD map in a fully supervised fashion. Here, the method is developed to make inference on the same data on which it is learnt without taking into account possible shifts in the underlying data distribution between training and test data.

Yang et al. (2019)proposed a DL-based CD method especially tailored to transfer a CD model from a source to a target domain. The proposed method involves two stages: i) a pre-training step in which the model is trained on the label abundant source domain and ii) a refinement step in which the model is fine-tuned according to pseudo-labels generated on the target domain in a self-training manner. The refinement stage exploits pseudo CD maps generated on the target data on which spatial reasoning, at region- and boundary-scale, is deployed to select target samples with associated pseudo-labels.

Shi, Liu, Li, Liu, Wang & Zhang (2022) designed and evaluated a deeply supervised attention metric-based network. CD maps are learnt by means of Siamese networks, while convolutional attention blocks are integrated with the aim to provide highly discriminative features. In addition, supervision is employed to enhance the feature extractor's learning ability and generate more useful features that are subsequently used to discriminate between changed and unchanged areas.

Finally, Ruzicka et al. (2020) explored AL in the context of neural network based remote sensing CD. The proposed work evaluates AL in a scenario characterised by a reduced amount of labelled source data to train the CD model. The method leverages as backbone model a Siamese network with an encoder pre-trained on the Imagenet dataset. To implement the AL process, the uncertainty related to an ensemble of Siamese network models is exploited as a criterion to sample new labelled data thus enriching the training set. While the inference is performed at pixel level, the method selects new samples at tile level thus, possibly introducing noisy information in the training data. The findings of this study highlight that the AL process permits to automatically balance the training distribution reaching out similar performances as a model supervised with a large pre-annotated training set. While this method shares with our proposal the idea to use AL, it differs from SENECA on two main aspects: firstly, the AL sampling strategy is purely based on model uncertainty without taking into account the spatial dimension that strongly characterizes remote sensing data and, secondly, it integrates new samples at tile level (patch of 256×256 pixels) thus introducing possible noisy labels conversely to our method in which pixel-level samples are integrated.

To sum up, the majority of CD methods only use the information from the current images themselves without taking into account possible distribution shifts between the training data (here referred as source domain) and the test data (here referred as target domain) that can negatively impact the final detection performances. When methods are proposed to explicitly manage such a data distribution shift, they mainly rely on heuristics (Shi, Liu, Li, Liu, Wang & Zhang 2022), sample selection based on uncertainty derived by the model output (Ruzicka et al. 2020) or self-training approaches that can introduce issues related to confirmation bias (Tarvainen & Valpola 2017) as well as mistakes due to large gaps between the source and the target domains.

3. Basics

Let us consider a MultiSpectral (MS) sensor technology (e.g., Sentinel-2) to observe the Earth's surface over K spectral bands. Every spectral band is a numeric feature proportional to the ultraviolet and short wavelength infrared for a given band. Let **scene** be a geographic scene spanned over $m_{\mathbf{scene}} \times n_{\mathbf{scene}}$ pixels, where a pixel denotes an area of around a few square meters of the Earth's surface (i.e., it is a function of the sensor's spatial resolution), which is unequivocally referenced with spatial coordinates (i, j), with $1 \leq i \leq m_{\mathbf{scene}}$ and $1 \leq j \leq n_{\mathbf{scene}}$, according to the usual matrix representation. A bi-temporal MS dataset $\mathbf{D}_{\mathbf{scene}}$ is composed of two co-registered MS images, i.e., $\mathbf{D}_{\mathbf{scene}} = (\mathbf{X}_{\mathbf{scene}}{}^1, \mathbf{X}_{\mathbf{scene}}{}^2)$, which describe MS data of **scene** acquired by using the Sentinel-2 MS sensor technology. Note that $\mathbf{X}_{\mathbf{scene}}{}^1$ and $\mathbf{X}_{\mathbf{scene}}{}^2$ are acquired in two distinct time periods, denoted as t^1 and t^2, respectively, with $t^1 < t^2$. Every MS image of $\mathbf{D}_{\mathbf{scene}}$ is represented as a tensor of $m_{\mathbf{scene}} \times n_{\mathbf{scene}}$ pixels and K spectral bands. For each dataset, the pixel indexed by row i and column j contains a vector of data sensed on that resolution cell over K spectral bands (MS vector). The pair $(\mathbf{X}_{\mathbf{scene}}{}^1(i, j), \mathbf{X}_{\mathbf{scene}}{}^2(i, j))$ denotes the bi-temporal MS vectors of $\mathbf{D}_{\mathbf{scene}}$ associated with pixel (i, j). Finally, the CD map $\mathbf{Y}_{\mathbf{scene}}$ of a bi-temporal dataset $\mathbf{D}_{\mathbf{scene}}$ is a matrix of $m_{\mathbf{scene}} \times n_{\mathbf{scene}}$ binary labels with $\mathbf{Y}(i, j) = 1$ if a change occurred in the surface covered by pixel (i, j) from t^1 to t^2; 0 otherwise.

4. Proposed Method

We assume that a MS sensor technology with K MS bands is used to monitor both a source scene $\mathbf{S}$ and a target scene $\mathbf{T}$, respectively. Both $\mathbf{S}$ and $\mathbf{T}$ covering different geographical areas. Each MS image of scene $\mathbf{S}$ is represented as $m_{\mathbf{S}} \times n_{\mathbf{S}}$ pixels and K spectral bands. Each MS image of scene $\mathbf{T}$ is represented as $m_{\mathbf{T}} \times n_{\mathbf{T}}$ pixels and K spectral bands. Let us consider: (1) a bi-temporal MS dataset $\mathbf{D}_{\mathbf{S}} = (\mathbf{X}_{\mathbf{S}}{}^1, \mathbf{X}_{\mathbf{S}}{}^2)$ of $\mathbf{S}$; (2) the ground truth CD map $\mathbf{Y}_{\mathbf{S}}$ of $\mathbf{D}_{\mathbf{S}}$; and (3) a bi-temporal MS dataset $\mathbf{D}_{\mathbf{T}} = (\mathbf{X}_{\mathbf{T}}{}^1, \mathbf{X}_{\mathbf{T}}{}^2)$ of $\mathbf{T}$. The CD methodology of SENECA, schematised in Figure 1, is mainly based on four components:

- The training of a CD model (pre-trained source CD model) from the labelled source, bi-temporal MS dataset.
- The use of a segmentation-based AL strategy to divide the target scene in super-pixel objects, select the medoids of the super-pixel objects and

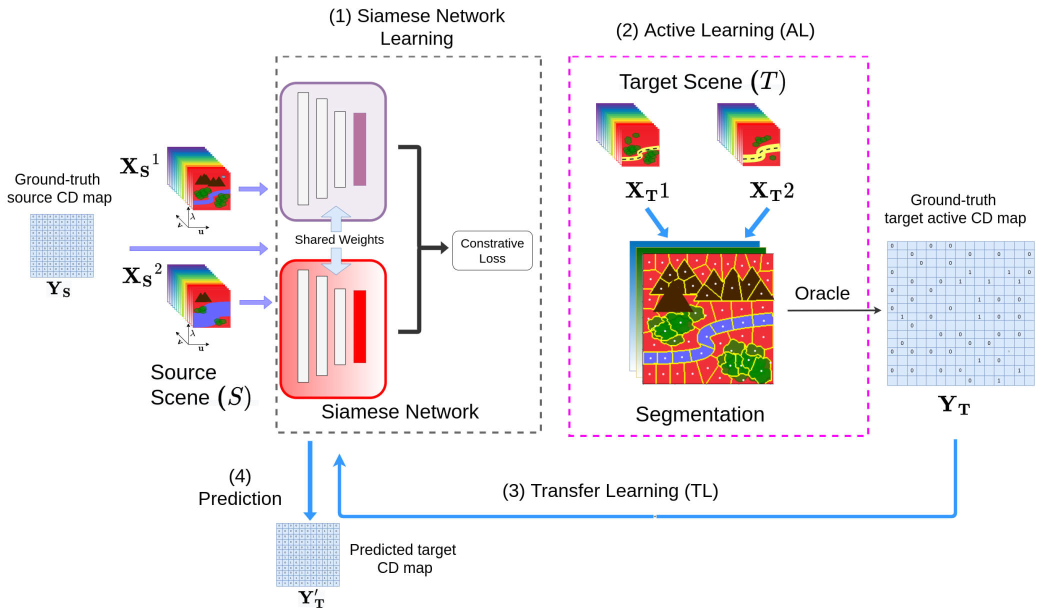


Figure 1: Schema of SENECA: (1) A CD Siamese network is trained from the pair of bi-temporal images $\mathbf{X_S}^1$ and $\mathbf{X_S}^2$ of a source scene $\mathbf{S}$ and the ground truth CD map $\mathbf{Y_S}$. (2) A segmentation-based AL strategy is used to select samples of bi-temporal images $\mathbf{X_T}^1$ and $\mathbf{X_T}^2$ of a target scene $\mathbf{T}$ and acquire their CD labels $\mathbf{Y_T}$. (3) A fine tuning-based TL strategy is used to update the parameters of the source Siamese network model with the limited samples of the target bi-temporal images $\mathbf{X_T}^1$ and $\mathbf{X_T}^2$ for which the CD labels $\mathbf{Y_T}$ have been acquired through the AL strategy. (4) The fine-tuned target CD Siamese network is used to predict the still unknown labels of the CD map for the target images and build the complete CD map $\mathbf{Y'_T}$.

acquire the labels of the pixel medoids selected through the segmentation step.

- The use of a fine tuning-based TL strategy to update the parameters of the source CD model with limited MS data of the target, bi-temporal MS dataset for which the labels have been acquired through the AL strategy (target CD model).

- The use of the target CD model, updated with fine tuning, to predict the still unknown labels of the CD map of the target, bi-temporal MS dataset.

A detailed description of the four components is reported in the following.

Table 1: List of used symbols

Symbol	Meaning
$\mathbf{S}$	Source scene of $m_{\mathbf{S}} \times n_{\mathbf{S}}$ pixels
$\mathbf{T}$	Target scene of $m_{\mathbf{T}} \times n_{\mathbf{T}}$ pixels
$\mathbf{D_S}$	Source bi-temporal MS dataset composed of co-registered MS images ${\mathbf{X_S}}^1$ and ${\mathbf{X_S}}^2$ of the scene $\mathbf{S}$
$\mathbf{D_T}$	Target bi-temporal MS dataset composed of co-registered MS images ${\mathbf{X_T}}^1$ and ${\mathbf{X_T}}^2$ of the scene $\mathbf{T}$
$\mathbf{Y_S}$	Ground-truth source CD map
$\mathbf{A_T}$	Target active scene
$\mathbf{Y_T}$	Ground-truth target active CD map
${\mathbf{Y_T}}'$	Predicted target CD map
$f_{\mathbf{S}}(\cdot)$	Embedding learned with the source Siamese network
$f_{\mathbf{T}}(\cdots)$	Embedding learned with the target Siamese network
θ	Otsu's threshold

The list of used symbols is reported in Table 1.

4.1. Source Siamese network

A Siamese network is pre-trained as a source CD deep neural model. In particular, the source CD Siamese network is trained by minimising a loss function computed on the sample distance of all the $m_{\mathbf{S}} \times n_{\mathbf{S}}$ bi-temporal MS vectors $({\mathbf{X_S}}^1(i,j), {\mathbf{X_S}}^2(i,j)) \in \mathbf{D_S}$ having labels $\mathbf{Y_S}(i,j) \in \mathbf{Y_S}$.

The Siamese network architecture consists of two identical supervised neural networks with shared weights, in order to learn the hidden representation (embedding) of the bi-temporal, MS vectors recorded in $\mathbf{D_S}$. The two neural networks are both feed-forward multi-layer perceptrons, and employ error back-propagation during training. They work in parallel comparing the embedding outputs at the end through Euclidean distance. With the aim to learn the model weights, we use the contrastive loss function that was originally proposed by Hadsell et al. (2006) to minimise the Euclidean distance, in the embedding space, between two samples that belong to the same class label and maximises the distance between two samples with different labels. In SENECA, all the bi-temporal MS vectors $({\mathbf{X_S}}^1(i,j), {\mathbf{X_S}}^2(i,j)) \in \mathbf{D_S}$ that are labelled with the class $\mathbf{Y_S}(i,j) = 1$ (*change*) are handled as pairs of samples with different land cover, while all the bi-temporal MS vectors that are labelled with the class 0 (*non-change*) are handled as pairs of samples labelled with the same land cover. Hence, the contrastive loss is defined as follows:

$$\mathcal{L}_c = \sum_{i,j \in \mathbf{S}} \left((1 - \mathbf{Y_S}(i,j))d_{\mathbf{S}}(i,j)^2 + \mathbf{Y_S}(i,j)\max\left(\alpha - d_{\mathbf{S}}(i,j), 0\right)^2\right), \quad (1)$$

where $f_{\mathbf{S}}(\cdot)$ is the embedding learned with the source Siamese network, $d_{\mathbf{S}}(i,j) = ||f_{\mathbf{S}}(\mathbf{X_S}^1(i,j)) - f_{\mathbf{S}}(\mathbf{X_S}^2(i,j))||_2$ and α is the margin. Notice that, during this training stage, the desired source embedding $f_{\mathbf{S}}(\cdot)$ is learned achieving that the distance between the bi-temporal MS vectors of the changed pixels of $(\mathbf{D_S}, \mathbf{Y_S})$ get larger than the unchanged pixel distances of $(\mathbf{D_S}, \mathbf{Y_S})$ by a margin of α.

4.2. Active learning

An AL strategy is used to identify a portion of the target scene $\mathbf{A_T} \subseteq \mathbf{T}$ (active target scene) that covers few relevant pixels of $\mathbf{T}$ (active pixels) for which it is suitable to acquire the unknown CD labels associated with the bi-temporal MS vectors contained in $\mathbf{D_T}$. To this aim, a segmentation algorithm is used, in order to group together similar adjacent pixels in visually meaningful spatial regions – super-pixel objects – that can be used to reduce the number of primitives for the AL analysis. In this study, we use the Simple Linear Iterative Clustering algorithm, referred as SLIC (Achanta et al. 2012), as segmentation approach. SLIC is inspired by the standard k-means clustering algorithm, in order to generate super-pixel object. The complexity of SLIC is linear in the number of pixels and independent of the number of super-pixels. It adopts a weighted distance measure combines colour and spatial proximity while simultaneously providing control over the size and compactness of super-pixel objects. In SENECA, the segmentation is performed to divide the target scene $\mathbf{T}$ into super-pixel objects and, successively, sample an active pixel to label for each super-pixel object. This segmentation step is expected to allow the selection of active pixels for both classes (*change*=1 and *non-change*=0) in $\mathbf{T}$ based on the MS information enclosed in $\mathbf{D_T}$.

To perform the segmentation step, first the tensor $\mathbf{X_T} = \mathbf{X_T}^1 \bullet \mathbf{X_T}^2$ is built by applying pixel-wise the concatenation operator $\bullet$ through the MS bands of both ${\mathbf{X_T}}^1$ and ${\mathbf{X_T}}^2$. More precisely, $\mathbf{X_T}$ is a tensor of $m_\mathbf{T} \times n_\mathbf{T}$ pixels and $2K$ spectral bands. The spectral dimensionality of $\mathbf{X_T}$ is then reduced from $2K$ bands to 2 principal components – PC_1 and PC_2. This preprocessing step is based on a previous study (Deng et al. 2008) that used the Principal Component Analysis (PCA) as transformation to better highlight the difference between two images. Based upon the theory reported in this previous study, the change can be identified in the second component, while the first component is assumed to be the sum of the common information. Subsequently, SLIC is used to segment $\mathbf{T}$ into κ super-pixel objects based on the information enclosed in $\mathbf{X^T}$. The user-defined parameter κ allows us to control the number of super-pixel objects and, therefore, the number of active exemplars sampled through the super-pixel objects. In particular, for each super-pixel object $\mathbf{o}$, the medoid pixel of $\mathbf{o}$, i.e., the pixel of $\mathbf{o}$ that is the closest in space to the centre of $\mathbf{o}$, is identified. Formally,

$$medoid(\mathbf{o}) = \underset{(i,j)\in\mathbf{o}}{\operatorname{argmin}} \left((i-i_c)^2 + (j-j_c)^2\right), \tag{2}$$

where (i_c, j_j) is the centre of $\mathbf{o}$ having coordinates $i_c = \frac{\sum\limits_{(i,j)\in\mathbf{o}} i}{|\mathbf{o}|}$ and $j_c = \frac{\sum\limits_{(i,j)\in\mathbf{o}} j}{|\mathbf{o}|}$. Finally, the active target scene $\mathbf{A_T}$ is populated with the medoid pixels of the super-pixel objects:

$$\mathbf{A_T} = \{medoid(\mathbf{o}) | \mathbf{o} \in SLIC(\mathbf{X^T})\}. \tag{3}$$

Notice that the active target scene $\mathbf{A_T}$ defines the AL-based set of relevant pixel exemplars of $\mathbf{T}$ whose ground truth CD labels $\mathbf{Y_T}$ are acquired with respect to the bi-temporal MS vectors of $\mathbf{D_T}$. $\mathbf{A_T}$ is used to complete the limited supervision of the target Siamese network with the TL strategy. Henceforth, we rely on $\mathbf{Y_T}(i,j) = 0/1$ for each $(i,j) \in \mathbf{A_T}$, *unknown* otherwise.

4.3. Transfer learning

A TL strategy is used to adapt the embedding $f_\mathbf{S}(\cdot)$ pre-trained on $\mathbf{D_S}$ to $\mathbf{D_T}$. This adaptation is completed using the limited supervision provided by the labels acquired in $\mathbf{Y_T}$ in correspondence of active pixels of $\mathbf{A_T}$. In particular, the fine tuning strategy is applied. This is an application of the transfer learning principle in deep learning (Tan et al. 2018) that allows us to train a deep neural model using limited labelled samples of a target distribution. Instead of weights being randomly initialised, they are those pre-trained on samples from a different – but related – source distribution. In this study, the fine tuning strategy starts with the weights of the pre-trained Siamese network that has learned the source embedding $f_{\mathbf{S}}$. Subsequently, it updates these weights to minimise the contrastive loss formulated in Eq. 1 and right now evaluated on the bi-temporal MS vectors of $\mathbf{D_T}$ and the labels of $\mathbf{Y_T}$, which are associated with active pixels of $\mathbf{A_T}$. This allows us to adapt the pre-trained Siamese network to new changes in the target bi-temporal MS dataset without retraining from scratch with limited class estimates only, which would incur in significant overhead and cause artefacts. We denote $f_{\mathbf{T}}(\cdot)$ the target embedding trained with the fine tuning strategy.

4.4. Target CD map

Finally, $f_{\mathbf{T}}(\cdot)$ is used to predict the unknown CD map $\mathbf{Y_T}'$ associated with $\mathbf{D_T}$. For each pixel $(i,j) \in \mathbf{A_T}$, $\mathbf{Y_T}'(i,j) = \mathbf{Y_T}(i,j)$, where $\mathbf{Y_T}(i,j)$ is the CD label acquired in the AL step. For each pixel $(i,j) \in \mathbf{T} - \mathbf{A_T}$, $\mathbf{Y_T}'(i,j)$ is predicted as follows:

$$\mathbf{Y_T}'(i,j) = \begin{cases} 1 & ||f_{\mathbf{T}}(\mathbf{X_T}^1(i,j)) - f_{\mathbf{T}}(\mathbf{X_T}^2(i,j))||_2 \geq \theta \\ 0 & otherwise \end{cases}. \tag{4}$$

In Eq. 4, the threshold θ is automatically identified with the Otsu's algorithm (Otsu 1972). This is an adaptive threshold algorithm that is commonly used in image binarization problems to turn a single intensity threshold that separates pixels into two classes. Using the Otsu's algorithm, the threshold is determined by minimising the intra-class intensity variance defined as a weighted sum of variances of the two classes [4]. To this aim, we assume that the MS bi-temporal vector distances, computed pixel-wise in $\mathbf{D_T}$, are represented in an histogram with L equal-width bins (levels) denoted as $[1, \ldots, L]$. Let η_i be the number of pixels at level i, so that $\sum_{i=1}^{L} \eta_i$ corresponds to the

[4] Minimising the intra-class variance is equivalent to maximising the inter-class variance, since the total variance (the sum of the intra-class variance and the inter-class variance) is constant for different partitions.

total number of pixels in the target scene T, i.e., $\sum_{i=1}^{L} \eta_i = n_{\mathbf{T}} m_{\mathbf{T}}$. According to this, the probability of each level i is computed as $p_i = \frac{\eta_i}{n_{\mathbf{T}} m_{\mathbf{T}}}$. The Otsu's algorithm identifies the optimal threshold level θ, in order to divide the pixels of the target scene into the class 0 (no-change), spanned over the distance levels $[1, 2, \ldots, \theta]$, and the class 1 (change), spanned over the distance levels $[\theta+1, \ldots, L]$, respectively. The optimal θ is chosen with the goal to minimize the intra-class variance that is defined as a weighted sum of variances of the two classes:

$$\theta = \underset{1 \leq \theta \leq L}{\operatorname{argmin}} \left(w_0(\theta) \sigma_1^2(\theta) + w_1(\theta) \sigma_2^2(\theta) \right), \tag{5}$$

where $\sigma_1^2(\theta)$ ad $\sigma_2^2(\theta)$ are the variance computed on the two classes separated by θ. Finally, the weights $w_0(\theta)$ and $w_1(\theta)$ are the probabilities of the two classes, which are computed as follows:

$$w_0(\theta) = \sum_{i=1}^{\theta} p_i \text{ and } w_1(\theta) = \sum_{i=\theta+1}^{L} p_i. \tag{6}$$

Final considerations concern the fact that the predicted CD map can contain errors or mistakes. To cope with these issues, we may apply the principle of local auto-correlation of objects, according to which detected clusters, comprising changed objects, generally expand across contiguous regions (Appice & Malerba 2019). Based on this principle, we may decide to change the assignment of pixels that strongly disagree with surrounding assignments. It mainly corresponds to performing a spatial-aware correction of the change assignment defined with Otsu's threshold. This correction, also used in (Appice et al. 2020, Andresini et al. 2022), assigns each pixel to the label that originally groups the majority of its neighbouring pixels reached within a fixed radius, in order to ensure spatial smoothness reducing salt and pepper errors.

4.5. Time complexity

The time complexity of SENECA is the sum of the time costs of training a Siamese Network, selecting active samples with the segmentation-based AL strategy and performing the fine-tuning of the Siamese Network on the active samples. The time cost of both training and fine-tuning a Siamese Network depends on the cost of training a deep neural network. This mainly depends on the cost of computing the gradient descent (in the back-propagation

Table 2: Characteristics (acquisition time points, scene size and percentage of changed pixels) of the bi-temporal MS images gathered with Sentinel-2 satellites in five scenes (**Abu Dhabi**, **Beihai**, **Beirut**, **Cupertino** and **Las Vegas**)

Scene	Timestamp 1	Timestamp 2	Scene size	%Change
Abu Dhabi	Jan 20, 2016	Mar 28, 2018	785×799	0.037
Beihai	Dec 09, 2016	Mar 09, 2018	772×902	0.024
Beirut	Aug 20, 2015	Oct 03, 2017	1070×1180	0.026
Cupertino	Sep 08, 2015	Mar 26, 2018	788×1015	0.023
Las Vegas	Aug 20, 2015	Feb 05, 2018	824×716	0.076

stage), that is, $\mathbf{O}(lwde)$, where l is the number of layers in the network, $w = \mathbf{O}(r^2)$ is the number of weights per layer, r is the maximum number of neurons per layer, d is the number of samples and e is the number of epochs. The time cost of the segmentation-based AL step mainly depends on the complexity of SLIC that is $\mathbf{O}(N)$ with N the number of segmented pixels.

5. Experimental Evaluation and Discussion

We evaluated the effectiveness of the CD methodology implemented by SENECA on five co-registered, bi-temporal MS images (see Section 5.1) that were acquired with Sentinel-2 satellites in locations picked all over both Asia and USA. The implementation of SENECA used in this evaluation is illustrated in Section 5.2. The measured performance metrics are described in Section 5.3, while the results are discussed in Section 5.4.

5.1. Imagery data description

We considered five co-registered, bi-temporal MS images[5] with various levels of visible urbanisation. Image were picked over Asia (**Abu Dhabi**, **Beihai** and **Beirut**) and USA (**Cupertino** and **Las Vegas**), respectively (Caye Daudt et al. 2019). Each image was gathered by the Sentinel-2 satellites of the Copernicus program, in 13 spectral bands between visible and short wavelength infrared in the period between 2015 and 2018. All bands

[5] `https://rcdaudt.github.io/oscd/`

are resampled at a spatial resolution of 10m. The pixel-level change ground truth was provided for each bi-temporal image with the annotated changes focused on urban land cover (e.g., new buildings or new roads). In this study, each bi-temporal MS imagery dataset was used for both learning a CD Siamese network with labelled data, as well as for fine tuning a pre-trained CD Siamese network with limited labelled samples. A summary of the characteristics of the bi-temporal MS images is reported in Table 2.

5.2. Implementation details

SENECA was implemented in Python 3.8, using Keras 2.4— a high-level neural network API with TensorFlow as the backend (Abadi et al. 2015). In the pre-processing step, the spectral data were scaled in the range $[0, 1]$ using the Min-Max normalization [6].

The Siamese network was implemented with two base feed-forward networks with shared weights. Each base network is a deep neural network with three layers with $256\times128\times64$ neurons and two dropout layers. The Rectified Linear Unit (ReLU) activation function was used as activation to each hidden layer and the contrastive function (Hadsell et al. 2006) was used as loss function. In the supervised initialization step, the weights were initialised following the Xavier scheme, while, in the fine tuning step, the weights saved from the previous network were used as a starting point. For each dataset, we optimized the hyper-parameter using the tree-structured Parzen estimator algorithm as implemented in the Hyperopt library (Bergstra et al. 2013). This hyper-parameter optimization was performed by using 20% of the entire training set as a validation set according to the Pareto Principle. We selected the hyper-parameter configuration that achieved the lowest validation loss. The hyper-parameters and their corresponding possible values are reported in Table 3. We trained the network with mini-batches using back-propagation, and the gradient-based optimization was performed using the Adam update rule (Kingma & Ba 2014).

For the AL strategy, we performed the segmentation step using the SLIC algorithm as implemented in Scikit-image library [7]. The number κ of segments to detect in the target scene through SLIC was set as a percentage

[6] `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html`

[7] `https://scikit-image.org/docs/dev/api/skimage.segmentation.html`

Table 3: Hyperparameter search space for the Siamese model.

Hyper-parameter	Values
batch size	$\{ 2^5, 2^6, 2^7, 2^8, 2^9 \}$
learning rate	[0.0001, 0.01]
dropout	[0,0.5]

$\kappa\%$ of the target scene size, where $\kappa\%$ is a user-defined parameter. By default $\kappa\% = 1\%$. Since SLIC algorithm processes RGB images, we scaled the two principal components extracted from the bi-temporal target dataset for the segmentation step in a range 0-255. In addition, we added a dummy component, set equal to 0, in order to create the third channel of the RGB representation.

Threshold θ used to predict the target CD map was estimated using the implementation of Otsu's algorithm from scikit-image library [8]. Finally, the radius of the kernel used for spatial-aware correction R was set equal to 5 for all the target scenes.

5.3. Performance metrics

In this Section we introduce the metrics measured to evaluate the accuracy of the predicted CD maps, the homogeneity of super-pixel segmentation and the efficiency of the learning process.

We measured the accuracy of the predicted CD maps with the Fscore (F1)(Tan et al. 2005), the Area Under the ROC curve (AUCROC)(Tan et al. 2005) and the Geometric mean (G-mean) (Kubat & Matwin 1997). These metrics are commonly considered in the remote sensing field for the evaluation of CD methods. Let us consider: tp – the number of pixels of the scene with the class *change* that are correctly predicted as belonging to that class type; fp – the number of pixels not belonging to the class *change* that are wrongly predicted as belonging to the class *change*; tn – the number of pixels not belonging to class *change* that are predicted as not belonging to class *change*; fn – the number of pixels of the class *change* that are wrongly predicted as not belonging to that class type; n is the total number of pixels in the scene. The F1 measures the harmonic mean of precision

[8]https://scikit-image.org/docs/dev/api/skimage.filters.html\#skimage.filters.threshold_otsu

and recall, i.e., $Fscore = 2\frac{precision \times recall}{precision + recall}$. The higher the F1, the better the balance between precision and recall achieved by the evaluated method. In particular, the precision measures how many pixels are correctly classified for the class *change*, given all predictions of that class type in the scene, i.e., $precision = \frac{tp}{tp+fp}$. The recall measures how many pixels are correctly predicted for the class *change* given all occurrences of that class type in the scene, i.e., $recall = \frac{tp}{tp+fn}$. The AUCROC measures the Area Under the ROC curve as it was defined with the False Positive Rate (FPR) on the x-axis and the True Positive Rate (TPR) on the y-axis. The FPR measures how many pixels are wrongly classified in the class *change* given all the occurrences of negative samples of that class type, i.e., $FPR = \frac{fp}{fp+tn}$. The TPR measures how many pixels are correctly predicted for the class *change* given all occurrences of that class, i.e., $TPR = \frac{tp}{tp+fn}$. Hence, the AUCROC value expresses the probability that a given method will rank a positive sample of the class *change* higher than a negative sample of the considered class. The G-mean measures the geometric mean of *specificity* and *recall* by equally considering the errors on both classes, i.e., $G - mean = \sqrt{specificity \times recall}$. In particular, the *specificity* measures how many pixels are correctly predicted for the class *unchange* given all occurrences of that class in the scene, i.e., $specificity = \frac{tn}{tn+fp}$.

In addition, we measured the homogeneity of the super-pixel objects with the Purity and F1. The Purity is a simple evaluation criterion of cluster quality. To compute Purity, each super-pixel object $\mathbf{o_i}$ identified through the segmentation step is assigned to the class c_j (*change* vs *non-change*) that is the most frequent in the object. The accuracy of this assignment is measured by counting the number of correctly assigned target pixels and dividing by the total number of target pixels $m_\mathbf{T} \times n_\mathbf{T}$, i.e., $Purity = \frac{1}{m_\mathbf{T} \times n_\mathbf{T}} \sum_{i=1}^{\kappa} \max_j |\mathbf{o_i} \cap c_j|$, where κ is the number of super-pixel objects detected in the segmentation step, while $|\cdot|$ denotes the cardinality operator. Similarly, the F1 of the segmentation output is measured by assuming the most frequent CD class observed in a super-pixel object as the CD class assigned by the segmentation to each pixel grouped in the super-pixel object. We measured F1 per class considering firstly the *change* class (F1 (*change*)) and, successively, the *non-change* class (F1 (*non-change*)). These two scores allow us to monitor the ability of the segmentation step of depicting super-pixel objects covering both homogeneous changed regions and homogeneous non-changed regions, respectively.

Finally, we evaluate the time performance (TIME) spent both learning the pre-trained CD model from the source scene and fine tuning a pre-trained CD model to the target scene. They were collected on a Linux machine with an Intel(R) Core(TM) i9-10900K CPU @ 3.70GHz and 64GB RAM. All the experiments are executed on a single GeForce RTX 3070. In this study, the training TIME was measured in minutes.

5.4. Results

The empirical validation was done with the Siamese network as CD model, in order to answer the following questions:

- Q1 To what extent the number of active pixels selected in the target scene by the AL labelling strategy has an effect on the performance of the CD model adapted with TL? (Sensitivity analysis in Section 5.4.1)
- Q2 Is the CD model adapted to a target scene with the proposed ATL strategy more powerful in labelling the target scene than the CD model pre-trained on the source scene? (Ablation study in Section 5.4.2)
- Q3 How does the performance of a CD model pre-trained on a source scene and adapted to a target scene through the proposed ATL strategy change with either the source scene or the target scene? (Source/target scene study in Section 5.4.3)
- Q4 Does the defined CD method outperform recent, state-of-the-art CD systems? (Competitor study in Section 5.4.4)

Experiments were performed by considering 20 configurations of source-target scenes. More precisely, for each target scene we generated four configurations by varying the source scene among the left-out scenes. For example, given the target scene **Abu Dhabi**, four configurations were generated by selecting the source scene among: **Beihai**, **Beirut**, **Cupertino** and **Las Vegas**, respectively.

5.4.1. Sensitivity analysis (Q1)

The sensitivity analysis was performed, in order to assess the influence of κ, i.e., the number of active pixels selected through the segmentation step on the behaviour of SENECA. As in the implementation of SENECA, $\kappa = \kappa\% \times n_{\mathbf{T}} \times m_{\mathbf{T}}$, we analysed the performance of SENECA by varying $\kappa\%$

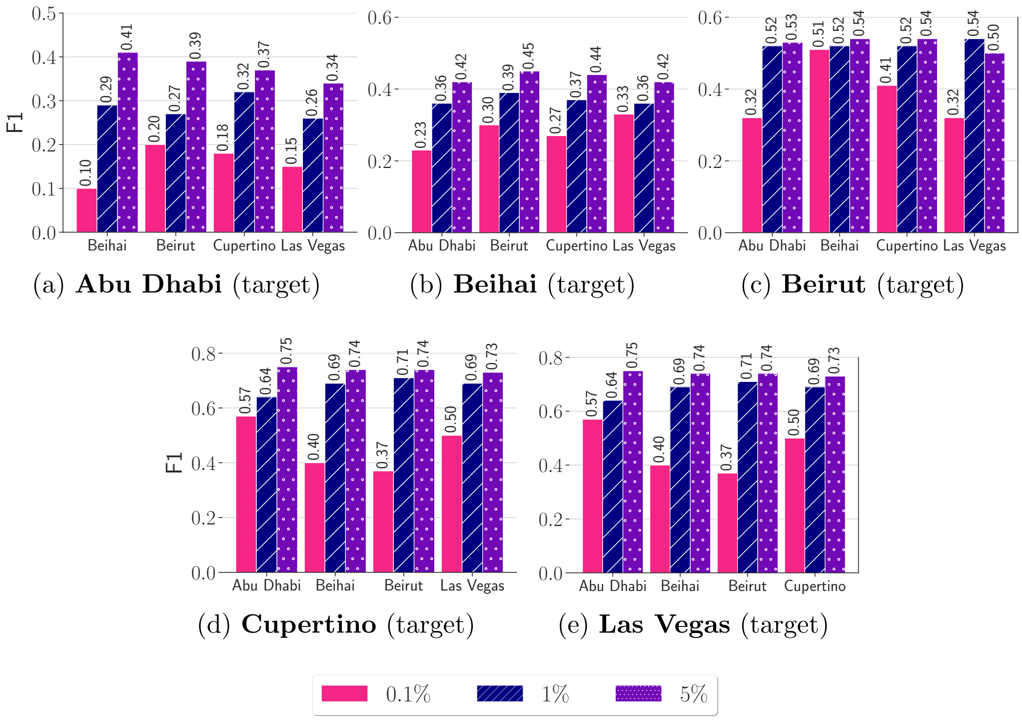
Table 4: F1, AUCROC, G-mean and TIME (in mins) of SENECA by varying $\kappa\%$ among $= 0.1\%$, 1% and 5%. We report the mean $\pm$ standard deviation of performances measured on all the target scenes with every CD model pre-trained with each left-out source scene.

$\kappa\%$	F1	AUCROC	G-mean	TIME
0.1%	0.40 (±0.20)	0.73 (±0.09)	0.68 (±0.10)	13.94 (±3.35)
1%	0.53 (±0.18)	0.76 (±0.05)	0.73 (±0.07)	92.92 (±48.97)
5%	0.57 (±0.17)	0.79 (±0.06)	0.76 (±0.09)	61040.86 (±2209.95)

among 0.1%, 1% and 5%. The mean and standard deviation of F1, AUCROC, G-mean and TIME measured for SENECA in all the tested configurations are reported in Table 4. Figure 2 reports the F1 computed for each target scene by varying both the source scene and $\kappa\%$. These results show that the higher the value of $\kappa\%$ (and, consequently, the higher the number κ of active pixels), the higher the accuracy of SENECA. On the other hand, this gain in accuracy is at the cost of the extra time spent completing the learning stage, as well as the higher effort and cost spent by experts acquiring the ground truth CD labels for the active pixels.

Additional conclusions can be drawn by analysing the homogeneity of super-pixel objects extracted through the segmentation step and considered to sample the active pixels of each scene. Figure 3 shows the segmentation's output of each considered scene as it was detected with $\kappa\% = 1\%$. Figure 4 reports the Purity, F1 for the class *change* and F1 for the class *non-change* as they were measured on the output of the segmentation step by varying $\kappa\%$ among 0.1%, 1% and 5%. These results reveal that the higher the value of $\kappa\%$, the finer-grained the segmentation of each scene in super-pixel objects and the higher the homogeneity of CD labels grouped in each super-pixel object. Detecting finer-grained super-pixel objects allows us to better depict homogeneous segments that mainly contain either changed pixels or non-changed pixels. Notably, the gain in the homogeneity of super-pixel objects is greater with respect to the class *change* than with respect to the class *non-change*.

In general, we note that $\kappa\% = 1\%$ allows us to achieve a good trade-off among homogeneity of segmentation, accuracy of final CD predictions, computation time spent completing the learning process, as well as effort



(a) **Abu Dhabi** (target) (b) **Beihai** (target) (c) **Beirut** (target)

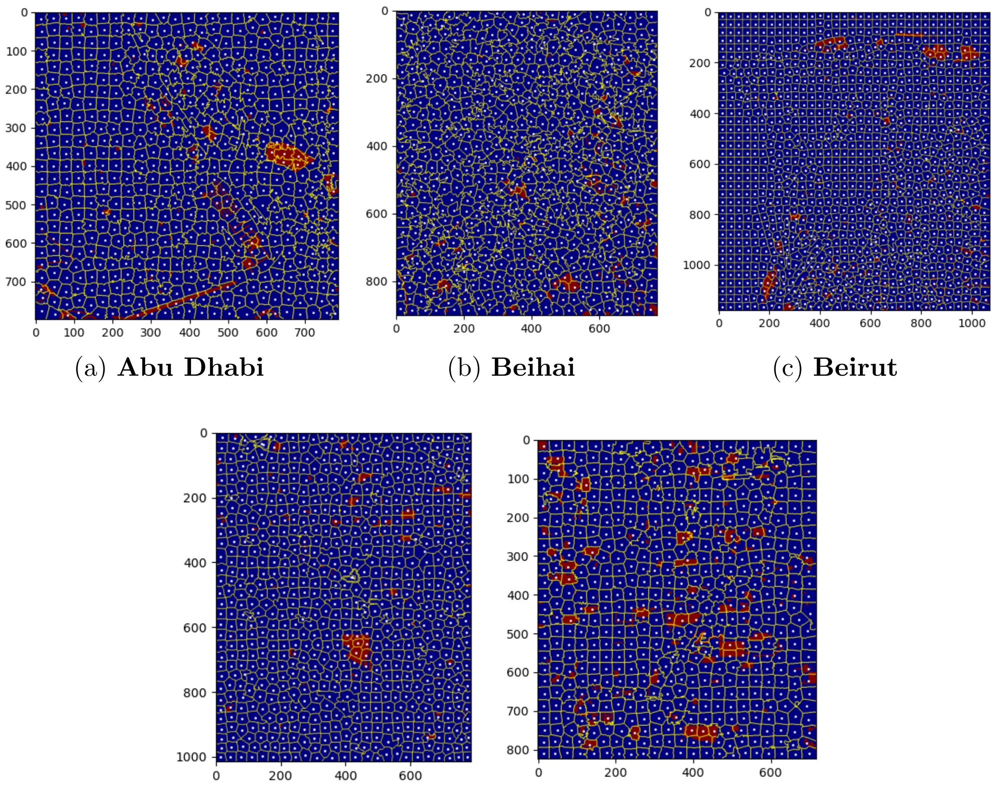
(d) **Cupertino** (target) (e) **Las Vegas** (target)

Figure 2: F1 of SENECA by varying $\kappa\%$ among 0.1%, 1% and 5%. For each target scene (Figures 2a-2e), we compare the F1 of the CD maps predicted by SENECA by varying the source scene.

and human cost spent acquiring CD labels. Due to these reasons, we report results achieved with $\kappa\% = 1\%$ in the rest of the experimental evaluation.

5.4.2. Ablation analysis (Q2)

The ablation analysis of SENECA was conducted, in order to explore how the ATL strategy can impact the performance of the CD model pre-trained on a source scene by adapting it to each left-out target scene. To this purpose, we ran the ATL strategy of SENECA with $\kappa\% = 1\%$ and measured the accuracy of the changes detected in each target scene by varying the source scene considered to learn the pre-trained CD model. For the ablation study, we also report the performance of Siamese that is the configuration that discards the ATL strategy. Specifically, Siamese used the CD model pre-trained on a source scene to detect changes of a target scene without performing any adaptation of the pre-trained CD model. The mean and standard deviations of F1, AUCROC, G-mean and TIME of both SENECA and Siamese are reported

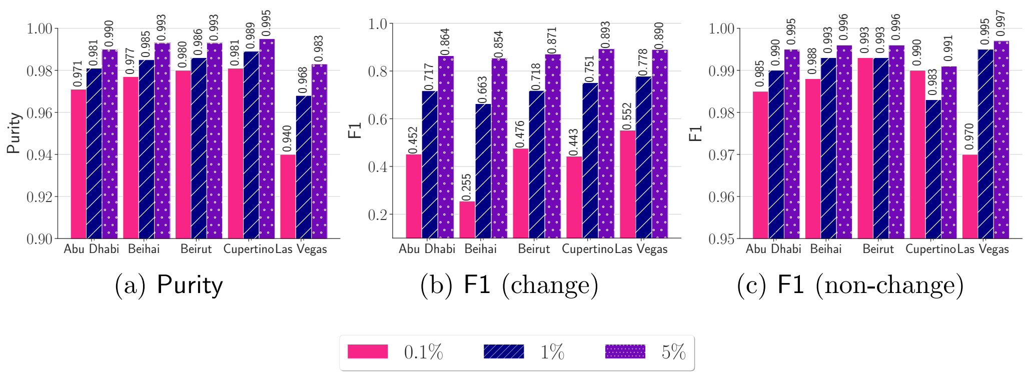


(a) **Abu Dhabi** (b) **Beihai** (c) **Beirut**

(d) **Cupertino** (e) **Las Vegas**

Figure 3: Super-pixel objects detected with the segmentation step performed with $\kappa\% = 0.1\%$. The red areas denote the changed regions, while the blue areas denote the unchanged regions in the corresponding scenes. White circles denote the active pixels sampled throughout the segmentation step.

in Table 5. Figure 5 reports the F1 scores computed per each target scene by varying the source scene. These results show that the use of the ATL strategy allows SENECA to gain accuracy compared to the baseline Siamese. Notably, this conclusion can be drawn independently of the source scene considered to train the source CD model. As expected, the higher accuracy of SENECA is at the cost of the more computation time spent performing the proposed ATL strategy. Figure 6 shows the computation time spent completing the four learning steps of SENECA in all the performed experiments. These results reveal that SENECA spends the most of its computation time performing the segmentation step in the AL component, while the time spent performing



(a) Purity (b) F1 (change) (c) F1 (non-change)

Figure 4: Purity, F1 of the class *change* and F1 of the class *non-change* measured on the output of the segmentation step performed by varying $\kappa\%$ among 0.1%, 1% and 5%

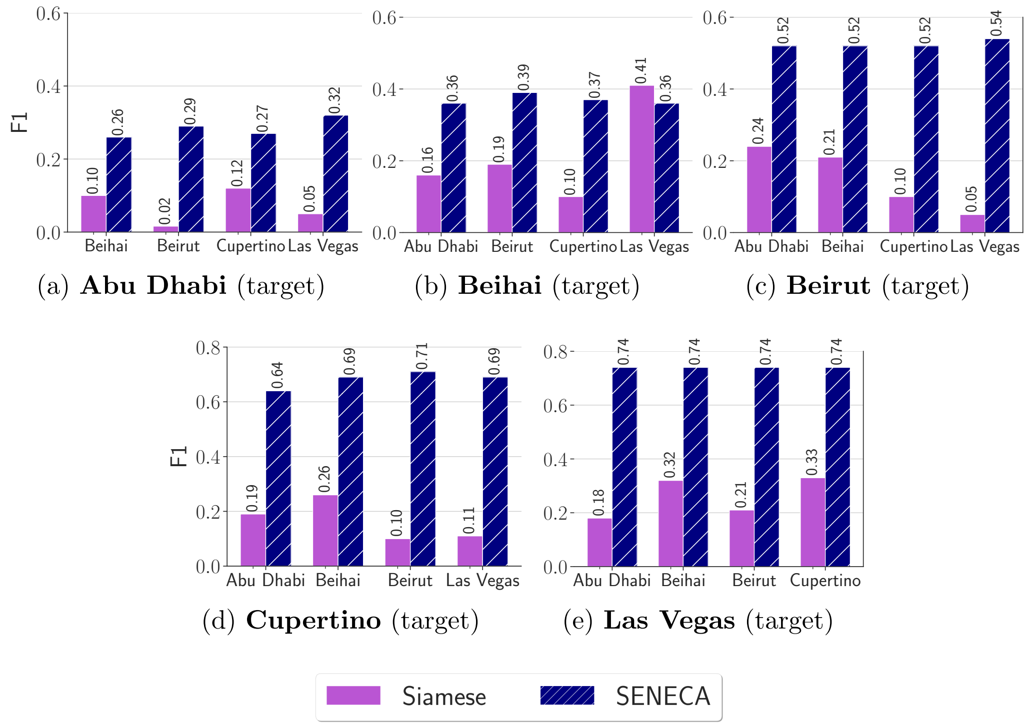
Table 5: F1, AUCROC, G-mean and TIME (in mins) of SENECA with $\kappa\% = 1\%$ and its baseline configuration Siamese. We report the mean $\pm$ standard deviation of performances measured on all the target scenes with every CD model pre-trained with each left-out source scene.

Method	F1	AUCROC	G-mean	TIME
SENECA	0.53 ($\pm$0.18)	0.76 ($\pm$0.05)	0.73 ($\pm$0.07)	92.92 ($\pm$48.97)
Siamese	0.18 ($\pm$0.09)	0.65 ($\pm$0.09)	0.61 ($\pm$0.11)	10.32 ($\pm$2.39)

fine tuning in the TL component is generally small.

5.4.3. Source and target scenes (Q3)

This analysis was conducted to explore the effect of a specific source/target scene on the accuracy of SENECA. Figure 7 shows the F1 of SENECA by varying both the source scene and the target scene. Results show that the accuracy performance of SENECA changes significantly with the target scene. However, the differences in the F1 of SENECA are commonly negligible in each target scene by varying the source scene. The only exception is observed with the target scene **Beihai** where the F1 varies from 0.36 (with the source scene **Abu Dhabi**) to 0.54 (with the source scene **Las Vegas**). Interestingly, also the source CD Siamese network pre-trained on **Las Vegas** outperforms the source CD Siamese network pre-trained on **Abu Dhabi**, **Beirut** and **Cupertino** when they were used to predict the CD map of **Beihai** without the



(a) **Abu Dhabi** (target) (b) **Beihai** (target) (c) **Beirut** (target)

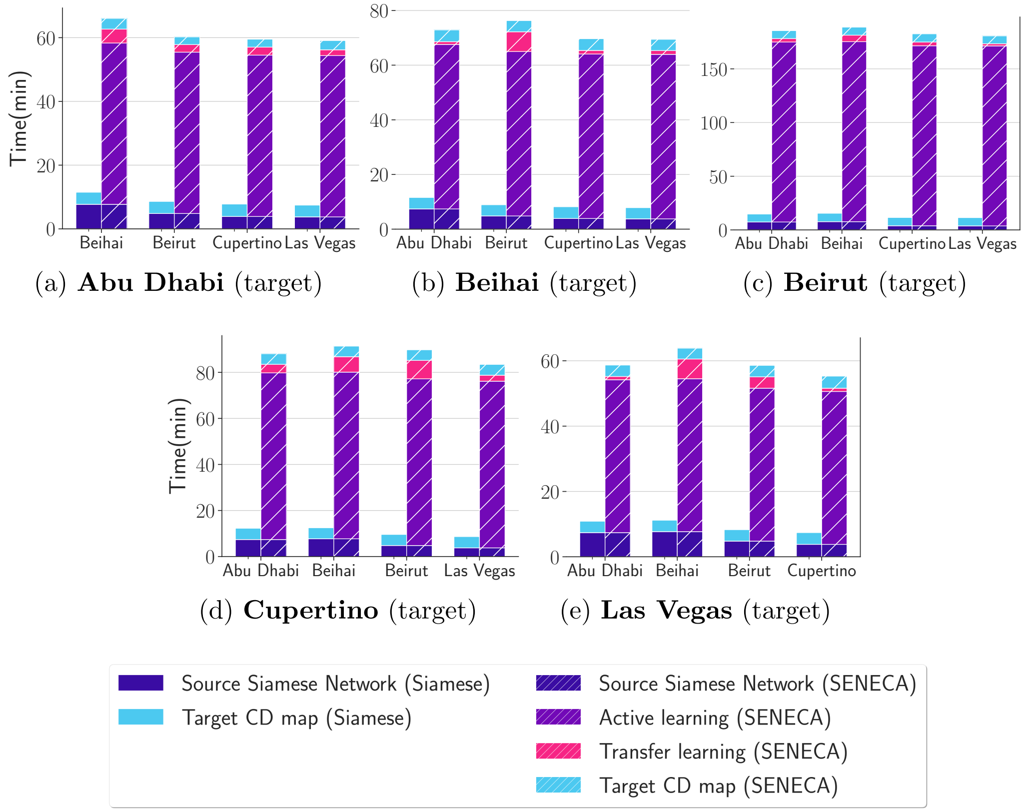
(d) **Cupertino** (target) (e) **Las Vegas** (target)

Figure 5: F1 of SENECA with $\kappa = 1\%$ and its baseline configuration Siamese. For each target scene (Figures 5a-5e), we compare the F1 of the CD maps predicted by both SENECA and Siamese by varying the source scene.

ATL strategy (see result of Siamese in Figure 5b). This suggests that a future research direction may focus on exploring which properties of the pre-trained CD models may foster the better performance of the ATL strategy.

5.4.4. Competitor analysis (Q4)

The comparative analysis is performed to assess the significance of accuracy and novelty of SENECA compared to several related methods, selected from the state of the art in CD literature. Table 6 reports a summary of the main characteristics of the considered competitors. We point out that the competitors that integrate the Siamese network (Shi, Liu, Li, Liu, Wang & Zhang 2022) and the ATL strategy (Ruzicka et al. 2020) are the closest to SENECA. Specifically the method CBAM described in (Shi, Liu, Li, Liu, Wang & Zhang 2022) introduces a convolutional attention block module in the Siamese network, but neglects any TL mechanism to adapt a CD model

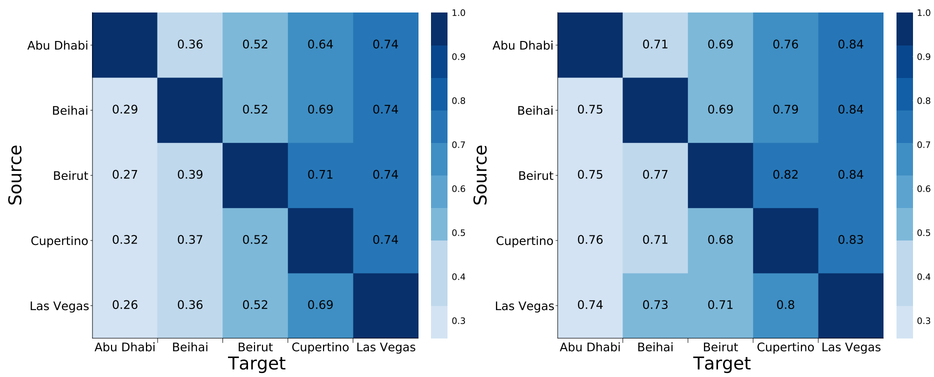


(a) **Abu Dhabi** (target) (b) **Beihai** (target) (c) **Beirut** (target)

(d) **Cupertino** (target) (e) **Las Vegas** (target)

Figure 6: TIME of SENECA with $\kappa = 1\%$ and its baseline configuration Siamese. For each target scene (Figures 6a-6e), we compare F1 of the CD map predicted by both SENECA and Siamese by varying the source scene.

trained in a source scene to a new target scene. The method SiameseU-Net described in (Ruzicka et al. 2020) trains a Siamese network with ResNet-34 base networks from a target source and uses an AL strategy to fine tune a source CD model to a target domain. In particular, it uses an ensemble procedure to select the tiles of pixels for the active labelling. It extends the source training set with the selected target active samples and re-trains the Siamese network from scratch using the augmented training set. In this comparative study, we experimented the AL strategy of both SiameseU-Net and SENECA to acquire the labels of the 1% of target samples. The methods BIC^2 , ORCHESTRA , PCAK-Means and CVA perform an unsupervised learning stage on the target scene by neglecting any information enclosed in the source scene. Finally, the method CBAM performs a supervised learning



(a) F1 (b) AUCROC

Figure 7: F1 of SENECA by varying both the target scene (axis X) and the source scene (axis Y)

Table 6: Compared algorithm description

Algorithm	**Description**
SENECA	Siamese network, ATL, Otsu's method
BIC^2(Appice et al. 2020)	GMM, PCA, Random Forest
ORCHESTRA(Andresini et al. 2022)	Autoencoder, CVA, spectral angle distance, Otsu'method
CBAM(Shi, Liu, Li, Liu, Wang & Zhang 2022)	Siamese network, ResNet18, Attention
SiameseU-Net(Ruzicka et al. 2020)	Siamese network, ResNet34, Active learning
PCAK-means(Celik 2009)	PCA, k-Means
CVA(López-Fandiño et al. 2019)	CVA, spectral angle distance, Otsu's algorithm

stage on the source scene and uses this pre-trained CD model on the target scene.

All the related methods were run using default parameters suggested by the authors in the reference papers. In particular, BIC^2 was run with the number of trees in the Random Forest set equal to 20, the number of principal components set equal to 20, the threshold considered to select the samples to train the Random Forests set equal to 0.85. Random Forests were constructed with the number of random independent features to look for the best split set equal to $\sqrt{\#independentfeatures}$, the bootstrap option was enabled with the bootstrap size set equal to the size of the training set and the function to measure the quality of a split, was set equal to the Gini index. The GMM was run with number of components set equal to 2, the covariance type set equal to diagonal (i.e. each component had its own diagonal covariance matrix), the non-negative regularisation added to the diagonal of covariance set equal to 0.00001. The image difference was computed with both the spectral angle distance and spectra-spatial cross correlation-based distance and the best results were considered for this comparative study.[9] ORCHESTRA was run with the autoencoder architecture composed of 3 fully-connected (FC) layers of $8 \times 4 \times 8$ neurons as proposed by the authors to process Sentinel-2 images. The learning rate and batch size were optimised with the tree-structured Parzen estimator in the range [0.00001, 0.01] and the set $\{32, 64, 128, 256, 512\}$, respectively. The optimisation was done using 20% of the entire training set as a validation set. The dropout layer was used to prevent overfitting. The mean squared error was used as the loss function. The ReLu was selected as the activation function for each hidden layer, while *Linear* activation function was used for the last layer. The number of epochs was set equal to 150, retaining the best models achieving the lowest loss on the validation set. CBAM was run with a Siamese Network implementing a ResNet18 (He et al. 2016) pre-trained on ImageNet. The ResNet18 was implemented with four basic blocks of depths equal to 64, 128, 256, and 512, respectively. Each basic block was composed by two convolutional layers with a kernel size of $3x3$ and two batch normalization layers. The model was fine-tuned for 150 epochs using Adam optimizer, ReLu activation was selected for each hidden layer. SiameseU-Net was run with a Siamese Network composed of two autoencoders with shared weights. The architecture of the encoder was implemented with a ResNet34 (He et al. 2016) pre-trained on ImageNet with a kernel size of $3x3$. The model was fine-tuned for 100 epochs using Adam optimizer and sigmoid as activation function for each hidden layer. The number of models used for the ensemble-based AL strategy was set equal to 5. PCAK-Means was run with the number of eigenvector equal to 3 and the block size equal to 4. CVA was run with the number of levels set equal to 256 in the Otsu's algorithm.

The mean and standard deviations of F1, AUCROC, G-mean and TIME of both SENECA and related methods are reported in Table 7. These results show that SENECA is able to outperform all the related methods in this study in terms of F1. On the other hand, PCAK-Means outperforms SENECA in terms of AUCROC and G-mean, where SENECA is the runner-up. This is a consequence of the fact that PCAK-Means discovers a higher number of

[9]The spectra-spatial cross correlation-based distance outperformed spectral angle distance in all scenes with the exception of **Las Vegas**.

Table 7: F1, AUCROC, G-mean and TIME (in mins) of SENECA with $\kappa\% = 1\%$, as well as the related methods. We report the mean ± standard deviation of performances measured on all the target scenes with every CD model pre-trained with each left-out source scene.

Method	F1	AUCROC	G-mean	TIME
SENECA	0.53 (±0.18)	0.76 (±0.05)	0.73 (±0.07)	92.92 (±48.97)
BIC^2	0.40 (±0.29)	0.70 (±0.14)	0.63 (±0.17)	11.88 (±5.36)
ORCHESTRA	0.23 (±0.20)	0.66 (±0.14)	0.65 (±0.18)	57.94 (±24.17)
CBAM	0.06 (±0.04)	0.50 (±0.07)	0.18 (±0.13)	16.58 (±3.46)
SiameseU-Net	0.33 (±0.19)	0.65 (±0.08)	0.65 (±0.18)	114.04 (±0.39)
PCAK-Means	0.40 (±0.21)	0.79 (±0.04)	0.78 (±0.04)	1.22 (±0.07)
CVA	0.24 (±0.20)	0.70 (±0.18)	0.64 (±0.17)	1.40 (±0.04)

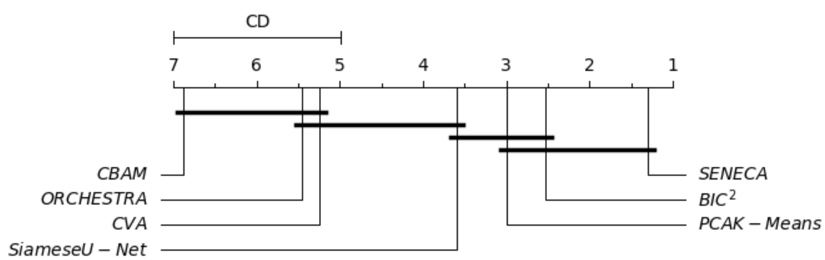


Figure 8: Nemenyi test of F1 of SENECA and related methods. Groups of methods that are not significantly different (at $p \leq 0.05$) are connected.

change samples (and consequently a lower number of $non - change$ samples) than SENECA. Hence, PCAK-Means performs a higher number of true positive samples, but also a higher number of false positive samples than SENECA. Therefore, SENECA outperforms PCAK-Means in terms of precision (0.58±0.29 in SENECA vs 0.34±0.26 in PCAK-Means) and specificity (0.97±0.04 in SENECA vs 0.91±0.08 in PCAK-Means), while PCAK-Means outperforms SENECA in terms of recall (0.55±0.11 in SENECA vs 0.67±0.06 in PCAK-Means). The impact of recall is higher in the formulation of G-mean and AUCROC than in the formulation of F1. This motivates differences in the observed performances of the compared methods with respect to F1, AUCROC and G-mean. In any case, a high number of false alarms (false positive) is not a desirable behaviour in imbalance classification problems such as CD

Table 8: F1 of SENECA ($\kappa\% = 1\%$), as well as the related methods. The best results are in bold.

Source	Target	SENECA	BIC^2	ORCHESTRA	CBAM	SiameseU-Net	PCAK-Means	CVA
Beihai	**Abu Dhabi**	**0.28**	0.19	0.15	0.07	0.24	0.19	0.17
Beirut		**0.27**	0.19	0.15	0.07	0.24	0.19	0.17
Cupertino		**0.31**	0.19	0.15	0.07	0.24	0.19	0.17
Las Vegas		**0.26**	0.19	0.15	0.07	0.24	0.19	0.17
Abu Dhabi	**Beihai**	**0.36**	0.09	0.05	0.09	0.34	0.41	0.05
Beirut		**0.39**	0.09	0.05	0.04	0.34	0.41	0.05
Cupertino		**0.36**	0.09	0.05	0.04	0.34	0.41	0.05
Las Vegas		**0.36**	0.09	0.05	0.04	0.34	0.41	0.05
Abu Dhabi	**Beirut**	**0.52**	0.35	0.06	0.05	0.08	0.20	0.06
Beihai		**0.51**	0.35	0.06	0.05	0.08	0.20	0.06
Cupertino		**0.51**	0.35	0.06	0.05	0.08	0.20	0.06
Las Vegas		**0.53**	0.35	0.06	0.05	0.08	0.20	0.06
Abu Dhabi	**Cupertino**	0.64	**0.68**	0.44	0.00	0.60	0.55	0.43
Beihai		**0.68**	**0.68**	0.44	0.04	0.60	0.55	0.43
Beirut		**0.70**	0.68	0.44	0.04	0.60	0.55	0.43
Las Vegas		**0.68**	**0.68**	0.44	0.03	0.60	0.55	0.43
Abu Dhabi	**Las Vegas**	**0.74**	0.72	0.46	0.00	0.41	0.67	0.45
Beihai		**0.73**	0.72	0.46	0.14	0.41	0.67	0.45
Beirut		**0.74**	0.72	0.46	0.15	0.41	0.67	0.45
Cupertino		**0.73**	0.72	0.46	0.04	0.41	0.67	0.45

tasks.

Further considerations concern the analysis of TIME. The DL-based methods (i.e., SENECA, ORCHESTRA, CBAM and SiameseU-Net) spent more time than the remaining methods (BIC^2, PCAK-Means and CVA). In any case, both SiameseU-Net and SENECA are the most time-consuming methods. Both methods train a Siamese network and integrate an AL-based strategy. However, SENECA uses a segmentation-based AL strategy, while SiameseU-Net uses an ensemble-based AL strategy.

We proceed this comparative study by examining in depth the F1 results per each scene. Results reported in Table 8 show that SENECA (with $\kappa\% = 1\%$) outperforms all the competitors of this study except for BIC^2 in the configuration with source **Abu Dhabi** and target **Cupertino**. However, SENECA outperforms (or performs equals to) BIC^2 on the target **Cupertino** when the source is **Beihai**, **Beirut** or **Las Vegas**. In addition, the highest accuracy on the target **Beirut** is achieved by SENECA with the source **Beihai**. Finally, we ranked the compared methods by statistically testing whether the improvement of F1 of the computed CD maps is significant over the various experimental configurations. To this aim, we have used Friedman's test (Demšar 2006). This is a non-parametric test that is commonly used to compare multiple methods over multiple experiments. It compares the average ranks of the methods, so that the best performing method gets the rank of 1. The second best gets rank 2. The null-hypothesis states that all the methods are equivalent. Under this hypothesis, the ranks of compared methods should be equal. In this study, we rejected the null hypothesis with *p-value* ≤ 0.05. As the null-hypothesis was rejected, that is, no method was singled out, we used a post-hoc test—the Nemenyi test—for pairwise comparisons (Demšar 2006). The results of this test reported in Figure 8 shows that SENECA enables the production of the CD map that commonly achieve the highest F1 by having BIC^2 as runner-up.

6. Conclusions

In this paper we have presented SENECA: an ATL methodology for CD in co-registered, bi-temporal MS images acquired with Sentinel-2 satellites in the same Earth's scene, at different time points. The proposed methodology uses the TL strategy to adapt the Siamese network pre-trained from a source domain to a related target domain. The adaptation is performed with the limited supervision made available with the AL strategy. An experimental study was performed to show the effectiveness of the proposed CD methodology, quantified in terms of CD accuracy. In particular, the results obtained have underlined that SENECA is able to produce decisions that outperform decisions produced with the baseline Siamese that is the configuration that discards the proposed ATL strategy. Furthermore, the experimental results clearly highlighted that SENECA achieves high quality performance with a limited amount of labels acquired via the AL process no matter the source data considered to learn the pre-trained Siamese network. Finally, the proposed ATL framework helps us to gain accuracy compared to various CD methods presented in the recent literature.

One limitation of the proposed methodology is the absence of any explanation mechanism. A future research direction could be devoted to explore eXplainable Artificial Intelligence mechanism, e.g., attentions or transformers, possibly coupled with convolutions, to get insights about particular spatial characteristics that may help to better recognise specific changes through a CD model. Another limitation is that the proposed methodology does not discriminate among different change types. This may be explored as multiclass ATL problem where new change classes may appear or disappear in the target domain with respect to the source domain. A further research direction refers to the systematic investigation of expected properties of both the source scene and target scene, to better foster the performance of the ATL strategy. Finally, recent studies have explored the CD problem in time series of co-registered MS images that exhibit some temporal trend in the change phenomena. Temporal change patterns may be explored to extend the proposed ATL strategy from the bi-temporal to the multi-temporal setting.

CRediT Authorship Contribution Statement

Giuseppina Andresini: Conceptualization, Methodology, Software, Data curation, Investigation, Validation, Visualization, Writing - original draft, Writing - review & editing. **Annalisa Appice**: Conceptualization, Methodology, Investigation, Validation, Supervision, Writing - original draft, Writing - review & editing. **Dino Ienco**: Conceptualization, Investigation, Writing - original draft, Writing - review & editing. **Donato Malerba**: Conceptualization, Writing - review & editing.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., ... & Zheng, X. (2015), 'TensorFlow: Large-scale machine learning on heterogeneous systems'.

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P. & Süsstrunk, S. (2012), 'Slic superpixels compared to state-of-the-art superpixel methods', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(11), 2274–2282.

Andresini, G., Appice, A., Iaia, D., Malerba, D., Taggio, N. & Aiello, A. (2022), 'Leveraging autoencoders in change vector analysis of optical satellite images', *J. Intell. Inf. Syst.* **58**(3), 433–452.

Appice, A., Guccione, P., Acciaro, E. & Malerba, D. (2020), 'Detecting salient regions in a bi-temporal hyperspectral scene by iterating clustering and classification', *Applied Intelligence* **50**(10), 3179–3200.

Appice, A. & Malerba, D. (2019), 'Segmentation-aided classification of hyperspectral data using spatial dependency of spectral bands', *ISPRS Journal of Photogrammetry and Remote Sensing* **147**, 215 – 231.

Bengio, Y., Courville, A. C. & Vincent, P. (2013), 'Representation learning: A review and new perspectives', *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(8), 1798–1828.

Bergstra, J., Yamins, D. & Cox, D. D. (2013), Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures, *in* 'ICML', pp. 115–123.

Caye Daudt, R., Le Saux, B., Boulch, A. & Gousseau, Y. (2019), 'OSCD - Onera Satellite Change Detection'.

Celik, T. (2009), 'Unsupervised change detection in satellite images using principal component analysis and k-means clustering', *IEEE Geoscience and Remote Sensing Letters* **6**(4), 772–776.

Daudt, R. C., Saux, B. L. & Boulch, A. (2018), Fully convolutional siamese networks for change detection, *in* '2018 IEEE International Conference on Image Processing, ICIP 2018, Athens, Greece, October 7-10, 2018', IEEE, pp. 4063–4067.

Demšar, J. (2006), 'Statistical comparisons of classifiers over multiple data sets', *J. Mach. Learn. Res.* **7**, 1–30.

Deng, J. S., Wang, K., Deng, Y. & Qi, G. J. (2008), 'Pca-based land-use change detection and analysis using multitemporal and multisensor satellite data', *International Journal of Remote Sensing* **29**(16), 4823–4838.

Gautheron, L., Habrard, A., Morvant, E. & Sebban, M. (2020), 'Metric learning from imbalanced data with generalization guarantees', *Pattern Recognition Letters* **133**, 298 – 304.

Hadsell, R., Chopra, S. & LeCun, Y. (2006), Dimensionality reduction by learning an invariant mapping, *in* '2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)', Vol. 2, pp. 1735–1742.

Hafner, S., Nascetti, A., Azizpour, H. & Ban, Y. (2022), 'Sentinel-1 and sentinel-2 data fusion for urban change detection using a dual stream u-net', *IEEE Geosci. Remote. Sens. Lett.* **19**, 1–5.

He, K., Zhang, X., Ren, S. & Sun, J. (2016), Deep residual learning for image recognition, *in* '2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)', pp. 770–778.

Jiang, H., Peng, M., Zhong, Y., Xie, H., Hao, Z., Lin, J., Ma, X. & Hu, X. (2022), 'A survey on deep learning-based change detection from high-resolution remote sensing images', *Remote. Sens.* **14**(7), 1552.

Kingma, D. P. & Ba, J. (2014), Adam: A method for stochastic optimization, *in* 'ICLR'.

Kubat, M. & Matwin, S. (1997), Addressing the curse of imbalanced training sets: One-sided selection, *in* 'In Proceedings of the Fourteenth International Conference on Machine Learning', Morgan Kaufmann, pp. 179–186.

Lewis, A., Lymburner, L., Purss, M. B. J., Brooke, B. P., Evans, B. J. K., Ip, A., Dekker, A. G., Irons, J. R., Minchin, S., Mueller, N., Oliver, S., Roberts, D. O., Ryan, B., Thankappan, M., Woodcock, R. & Wyborn, L. (2016), 'Rapid, high-resolution detection of environmental change over continental scales from satellite data - the earth observation data cube', *Int. J. Digit. Earth* **9**(1), 106–111.

Lu, J., Hu, J. & Zhou, J. (2017), 'Deep metric learning for visual understanding: An overview of recent advances', *IEEE Signal Processing Magazine* **34**(6), 76–84.

Lv, Z., Liu, T., Benediktsson, J. A. & Falco, N. (2022), 'Land cover change detection techniques: Very-high-resolution optical images: A review', *IEEE Geoscience and Remote Sensing Magazine* **10**(1), 44–63.

López-Fandiño, J., B. Heras, D., Argüello, F. & Dalla Mura, M. (2019), 'Gpu framework for change detection in multitemporal hyperspectral images', *Int J Parallel Prog* **47**, 272–292.

Ma, W., Wu, Y., Gong, M., Xiong, Y., Yang, H. & Hu, T. (2019), 'Change detection in sar images based on matrix factorisation and a bayes classifier', *International Journal of Remote Sensing* **40**(3), 1066–1091.

Otsu, N. (1972), 'A threshold selection method from gray-level histograms', *IEEE Trans. Geoscience and Remote Sensing* **9**(1), 62–66.

Ouali, Y., Hudelot, C. & Tami, M. (2020), 'An overview of deep semi-supervised learning', *CoRR* **abs/2006.05278**.

Pasolli, E., Melgani, F., Tuia, D., Pacifici, F. & Emery, W. J. (2014), 'SVM active learning approach for image classification using spatial information', *IEEE Trans. Geosci. Remote. Sens.* **52**(4), 2217–2233.

Ru, L., Du, B. & Wu, C. (2021), 'Multi-temporal scene classification and scene change detection with correlation based fusion', *IEEE Trans. Image Process.* **30**, 1382–1394.

Ruzicka, V., D'Aronco, S., Wegner, J. D. & Schindler, K. (2020), Deep active learning in remote sensing for data efficient change detection, *in* T. Corpetti, D. Ienco, R. Interdonato, M. Pham & S. Lefèvre, eds, 'Proceedings of MACLEAN: MAChine Learning for EArth ObservatioN Workshop co-located with the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD 2020), Virtual Conference, September 14-18, 2020', Vol. 2766 of *CEUR Workshop Proceedings*, CEUR-WS.org.

Shi, Q., Liu, M., Li, S., Liu, X., Wang, F. & Zhang, L. (2022), 'A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection', *IEEE Transactions on Geoscience and Remote Sensing* **60**, 1–16.

Shi, S., Zhong, Y., Zhao, J., Lv, P., Liu, Y. & Zhang, L. (2022), 'Land-use/land-cover change detection based on class-prior object-oriented conditional random field framework for high spatial resolution remote sensing imagery', *IEEE Trans. Geosci. Remote. Sens.* **60**, 1–16.

Sublime, J. & Kalinicheva, E. (2019), 'Automatic post-disaster damage mapping using deep-learning techniques for change detection: Case study of the tohoku tsunami', *Remote. Sens.* **11**(9), 1123.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C. & Liu, C. (2018), A survey on deep transfer learning, *in* 'Proc. of the International Conference on Artificial Neural Networks and Machine Learning (ICANN)'.

Tan, P.-N., Steinbach, M. & Kumar, V. (2005), *Introduction to Data Mining, (First Edition)*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Tarvainen, A. & Valpola, H. (2017), Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, *in* 'ICLR'.

Wu, Y., Bai, Z., Miao, Q., Ma, W., Yang, Y. & Gong, M. (2020), 'A classified adversarial network for multi-spectral remote sensing image change detection', *Remote Sensing* **12**(13).

Wu, Y., Li, J., Yuan, Y., Qin, A. K., Miao, Q.-G. & Gong, M.-G. (2021), 'Commonality autoencoder: Learning common features for change detection from heterogeneous images', *IEEE Transactions on Neural Networks and Learning Systems* pp. 1–14.

Yang, M., Jiao, L., Liu, F., Hou, B. & Yang, S. (2019), 'Transferred deep learning-based change detection in remote sensing images', *IEEE Trans. Geosci. Remote. Sens.* **57**(9), 6960–6973.