

QFabric: Multi-Task Change Detection Dataset

Sagar Verma,^{1,2} Akash Panigrahi,² Siddharth Gupta,²

¹ Université Paris-Saclay, CentraleSupélec, Inria, Centre de Vision Numérique

² Granular AI

sagar.verma@centralesupelec.fr, {sagar, akash, sid}@granular.ai

Abstract

Detecting change through multi-image, multi-date remote sensing is essential to developing and understanding of global conditions. Despite recent advancements in remote sensing realized through deep learning, novel methods for accurate multi-image change detection remain unrealized. Recently, several promising methods have been proposed to address this topic, but a paucity of publicly available data limits the methods that can be assessed. In particular, there exists limited work on categorizing the nature and status of change across an observation period.

This paper introduces the first labeled dataset available for such a task. We present an open-source change detection dataset, termed QFabric, with 450,000 change polygons annotated across 504 locations in 100 different cities covering a wide range of geographies and urban fabrics. QFabric is a temporal multi-task dataset with 6 change types and 9 change status classes. The geography and environment metadata around each polygon provides context that can be leveraged to build robust deep neural networks. We apply multiple benchmarks on our dataset for change detection, change type and status classification tasks. Project page: <https://sagarverma.github.io/qfabric>

1. Introduction

In order to address many of humanity’s biggest challenges it is imperative that we develop a thorough understanding of our planet and how it is evolving. Yet, there exist few sources for reliable, timely data on how the earth is changing and the impact humans are having on it. Without appropriate methods for monitoring and evaluation, it is incredibly difficult to measure the anthropogenic impact of an increasing global population.

Satellite data and geospatial machine learning offer an unparalleled source of objective global-scale data. The past decade has seen a near eight-fold increase in the number of earth observation satellites deployed to orbit. This increase represents a paradigm shift in the availability of satellite data and thus its potential in downstream applications. With

imagery now available for nearly every place on earth on a daily basis, researchers can now rely on satellites as a source for time-series data on natural and man-made changes. The ability to model earth’s land use and land cover (LULC), and how it is changing due to human activities and natural phenomena, would have manifold implications on the study of climate change, economic development and anthropology. Further, it would serve as a valuable tool to improve decision making in human aid and disaster relief efforts and provide critical insights into a variety of topics such as sustainable development and urban sprawl, water and air contamination levels, and illegal construction.

Yet, despite significant advances in machine vision over the past few years, geospatial change detection remains a challenging problem. Traditional methods, summarised in a survey by Radke et al.[24], use handcrafted techniques that heavily rely on environment dependent pre-processing and post-processing, making them ill-suited for large scale evaluation. Further, the accuracy that existing change detection methods demonstrate is heavily influenced by registration errors [30] and illumination changes [23] that do not correspond to true semantic changes. Such constraints limit their potential for use in operational settings.

Recently, a variety of change detection techniques have been proposed that cite advances in deep learning-based change detection methods from different fields [23, 32, 9, 10]. Yet a paucity of available training data hampers progress in this domain. Several change detection datasets have been introduced. Benedeck and Szirnyi [2] created the Aerial Change dataset which contain about 8 million labeled pixels, divided into three regions. Daudt *et al.* created the OSCD [9] dataset from Sentinel-2 multispectral images, with a total of about 9 million labelled pixels. Papadomanolaki *et al.* [23] extended this dataset from bi-date to multi-date to address problems associated with seasonal changes. High Resolution Semantic Change Detection (HRSCD) dataset [4] is the one of the largest change detection dataset. xBD [17] is a dataset for assessing building damages after disaster. SpaceNet [13] released a 24 month building tracking dataset [14].

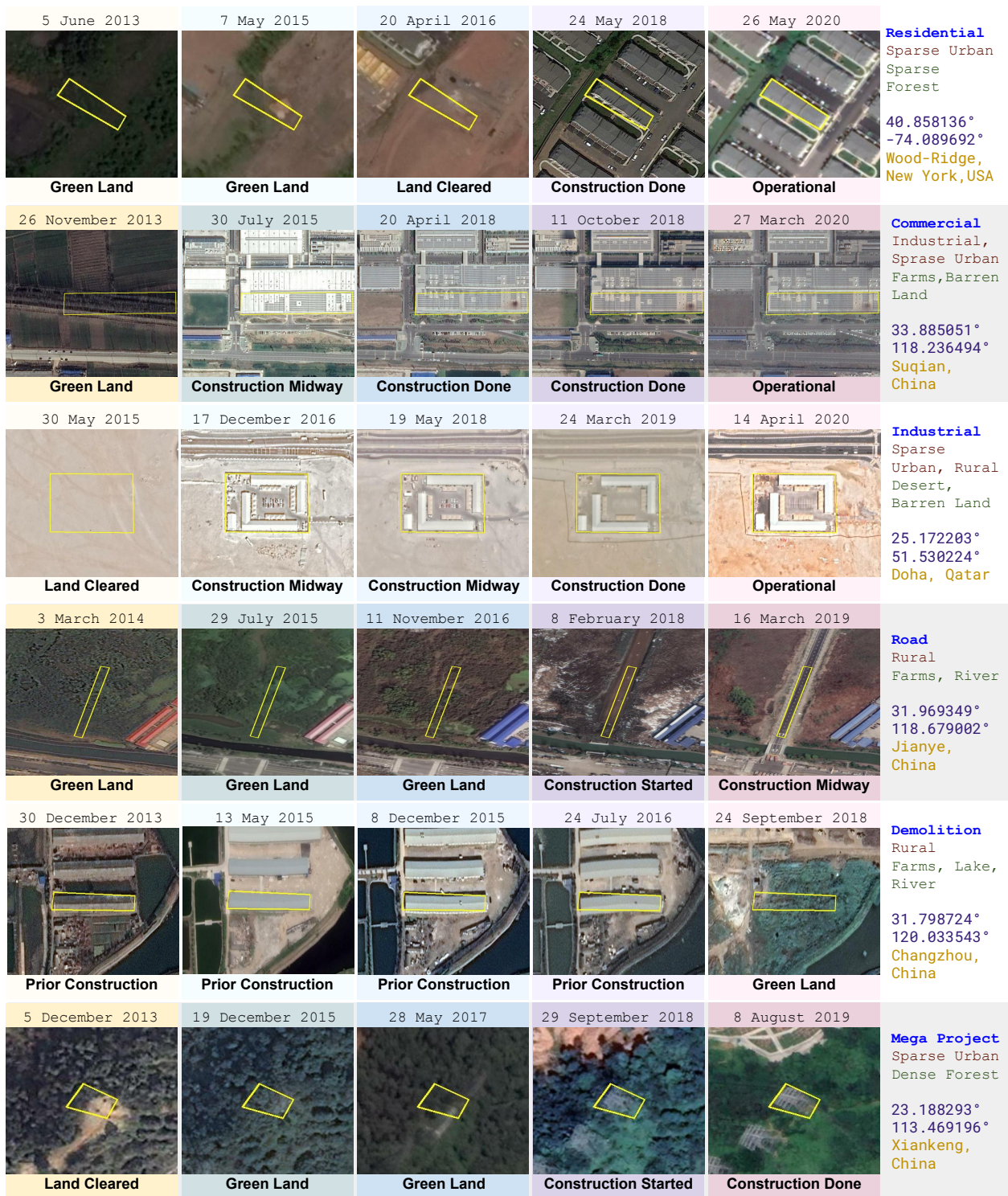


Figure 1. Sample polygons from QFabric showing different **change type**, **change status** on different dates, **neighborhood label(s)**, and **geography label(s)**. **Latitude-longitude** of the change polygon is shown along with **city name**. First row shows construction of a residential property in suburban area of New York, USA. Second row shows a commercial building in an industrial region which used to be farm lands of a fast growing second tier city in China. Third row shows an industrial construction in desert of Doha, Qatar which went from rural barren desert to a sparse urban area in a time period of 5 years. Fourth row shows construction of a road crossing a river in farm lands of rural China. Fifth row shows special case of urban change, demolition of a farm storage in the fast growing city Changzhou in China. Last row shows construction of a power grid unit which comes under mega project type.

Dataset	Gigapixels	#Images	Resolution (m)	#Objects	Annotation
OSCD [9]	9.6 Mega	48	10	1,769	Polygons
MultiDate OSCD [23]	48 Mega	120	10	1,769	Polygons
SpaceNet7 [14]	2.6	2,389	4	11,080,000	Polygons
HRSCD [11]	29.1	291	0.5	9,894	Polygons
QFabric (Ours)	245.1	2,520	0.31 - 0.70	450,000	Polygons + Classification
FMoW [8]	1,084.0	523,846	0.31-1.60	132,716	Classification
xView [19]	56.0	1,400	0.31	1,000,000	Bbox
SpaceNet [13] MVOI	50.2	60,000	0.46-1.67	126,747	Polygons
PascalVOC [15]	-	21,503	-	62,199	Bbox
MSCOCO [20]	-	123,287	-	886,266	Bbox
ImageNet [26]	-	349,319	-	478,806	Bbox

Table 1. Comparison of our dataset with other change detection , satellite classification and segmentation , and standard computer vision datasets.

2. Related Work

Change detection has a long history, being one of the early problems tackled in remote sensing image understanding [29]. It is done using coregistered image pairs or sequences, and consists of identifying areas in the images that have experienced significant modifications between the acquisitions. Many of the state-of-the-art methods for pattern recognition have been used for change detection in the past, from pixel-level comparison of images to superpixel segmentation, object-level image analysis, and image descriptors [18]. Traditional methods, summarised in surveys such as [24], use handcrafted techniques which are applied to images that cover large areas. Change detection is a non-trivial problem as the accuracy of a method is highly influenced by registration errors [30] and illumination changes that do not really correspond to semantic changes.

Recently, with the advancements in deep learning-based methods in different fields, a variety of change detection techniques have been proposed. In particular, in [21] a deep patch based architecture is proposed where bi-temporal patches are processed in parallel by a series of dilated convolutional layers generating features which are then fed to a recurrent sub-network to learn sequential information. In the end, fully connected layers are used to create the change prediction map. Although patch-based techniques produce promising results, they are time-consuming since they need to process every single pixel of the image individually. Most methods use transfer learning to overcome the problem of scarcity of training data through transfer learning by using pretrained networks to generate pixel descriptors [28, 1, 12]. Recently, Daudt *et al.* [10] suggested three different fully-convolutional Siamese networks based on the U-Net architecture [25] aiming to address this problem and detect accurately the regions with changes. However, this approach

is lacking the appropriate modeling of data’s temporal pattern. Several other methods that use CNN in Siamese settings are [33, 10, 9, 7, 16]. Papadomanolaki *et al.* [23] proposed a convolutional LSTM based approach that takes in five dates to detect changes in first and last date’s images. This method is resilient to seasonal changes, cloud covers and shadows. Daudt *et al.* [5] proposed a weakly supervised change detection method using guided anisotropic diffusion and iterative learning. [6] proposed a siamese-twins-based spatial-temporal attention network.

OSCD [9] dataset is a bi-date change detection dataset comprising of 24 Sentinel-2 image pairs of two dates from 24 different cities, mostly European. Seasonal change, cloud cover and shadows are present in some of the images. In total there are 1769 change polygons. Papadomanolaki *et al.* [23] extended OSCD dataset to 5 dates by introducing three intermediate dates. They show that by having more dates their proposed method is resilient to false positives occurring due to seasonal changes, cloud covers and shadows. LEVIR-CD [6] is a large-scale remote sensing building change detection dataset. LEVIR-CD consists of 637 very high resolution Google Earth image patch pairs with a size of 1024×1024 . These bitemporal images with time span of 5 to 14 years have significant land-use changes, especially the construction growth. LEVIR-CD covers various types of buildings, such as villa residences, tall apartments, small garages and large warehouses. Building change status from soil/grass/hardened ground to under construction to new build-up regions is monitored but only change and no-change labels are available. In total the dataset comprises of 31,333 individual change building instances. Optical Aerial Image dataset [2] are image pairs provided by Hungarian Institute of Geodesy Cartography & Remote Sensing and Google Earth. This dataset contains images taken in 2000 and 2005. Dataset part 1 consists

seven image pairs covering 9.5km-square areas at 1.5 meter/pixel resolution. One image pair is used for training and six are used for validation. Dataset part 2 consists of five image pairs from 2000 and 2007 covering 6.8 km-square area. Dataset part 3 consist of images taken from 1984 and 2007 from Google Earth. There is a huge resolution and exposure difference between these image pairs. HRSCD [11] contains 291 coregistered image pairs of RGB aerial images from IGS’s BD ORTHO database. Pixel-level change and land cover annotations are provided, generated by rasterizing Urban Atlas 2006, Urban Atlas 2012, and Urban Atlas 2006-2012 maps. 50cm resolution images are present in the dataset. Change and no-change labels are available along with six land cover classes for pixel level LULC. SpaceNet7 [14] is a open source dataset consisting of Planet satellite imagery at 4 meter pixel resolution. There are 24 images per month for 2 years covering 100 unique geographies. The dataset comprises over 40,000 square kilometers of imagery and exhaustive polygon labels of building footprints in the imagery, totalling over 10 million individual annotations.

3. Dataset Creation

In this section we describe the QFabric dataset in detail, including the categorisation, dates selection, and location selection criteria. We also elucidate our crowd-sourcing and annotation quality review process. Further, show relevant statistics about the dataset that describe the diversity, scale, and depth of the imagery.

QFabric contains imagery from 100 different cities collected by Maxar’s WorldView-2 Satellite between January 2014 and July 2020. We select 504 different locations from these 100 different cities maximizing dataset coverage of all geographic and urban types. For each of these 504 locations, we sample satellite images for five different dates such that duration between two consecutive date pairs are almost equal. An attempt was made to introduce different months to capture seasonal patterns and have different seasons present on different dates for each location. Image selection was further restricted to images with no more than 5% cloud cover.

3.1. Processing

Multi-view satellite imagery datasets are distinct from related natural image datasets in a number of interesting ways. First, as look angle increases in satellite imagery, the native resolution of the image decreases because greater distortion is required to project the image onto a flat grid. Second, each view contains images with multiple spectral bands. For our baseline, we used 3-channel images (RGB:red, green, blue), but also examined the contributions of the near-infrared (NIR) channel. These images were enhanced with a separate, higher resolution panchromatic (grayscale) channel to double the original resolution

of the multispectral imagery (i.e., ”pan-sharpened”). Each location is a tile of size 8192×8192 with a resolution of $0.45m \times 0.45m$ ground sample distance. The 16-bit pan-sharpened RGB-NIR pixel intensities were truncated at 3000 and then rescaled to an 8-bit range.

3.2. Annotations

Annotation is done in two stages. A team of 20 annotators are involved in annotation process. In the first stage, each location is assigned to an annotator to annotate all change polygons present in that location from first and last date images. In the second stage, each annotated polygon is used to crop out images from five date images. A buffer was introduced around the region-of-interest to ensure that annotators had a perspective on the surroundings, enabling them to annotate change type, change status, geographic type and urban type.

Quality control played an essential role in developing a rich multi-label dataset with complex and irregular polygons. At each stage of annotation, quality control processes were adapted to ensure that our annotations are consistent and valid. In the first stage of change polygons annotations, the quality reviewers selected 5 random 1024×1024 non-overlapping locations from a 8192×8192 location image and checked for wrong or missing annotations. The expert then calculated accuracy based on this location and communicated the accuracy back to the annotator along with detailed feedback. We observe that this strategy improves individual annotator’s accuracy quickly and we were able to get consistent and valid annotations with minimal iteration. In the second stage of labelling each change polygons, the expert randomly selected 30 images per class out of all the annotations done in a single day by an annotator. In total we have 31 class labels across four label types. The expert then checked whether these 30 images of a class are consistent and correct. Accuracy is calculated based on how many images are consistent and how many of them are not. This is communicated back to the annotator along with detailed feedback. If the accuracy is below 80% then all annotations of the annotator is discarded for that day and re-annotated.

All change polygons are provided in standard coordinate reference system, using the World Geodetic System 1984 (WGS 84) [WGS84] datum, with longitude and latitude units of decimal degrees. All change polygons for a single location along with associated four label types is embedded as GeoJSON [3]. For each change polygon we also provide neighborhood region bounding-box.

3.3. Dataset statistics

Our dataset labels comprises a broad distribution of change polygon sizes, as shown in Figure 2A. Compared to natural image datasets, our dataset more heavily emphasizes on small objects, with majority of objects less that 12,000

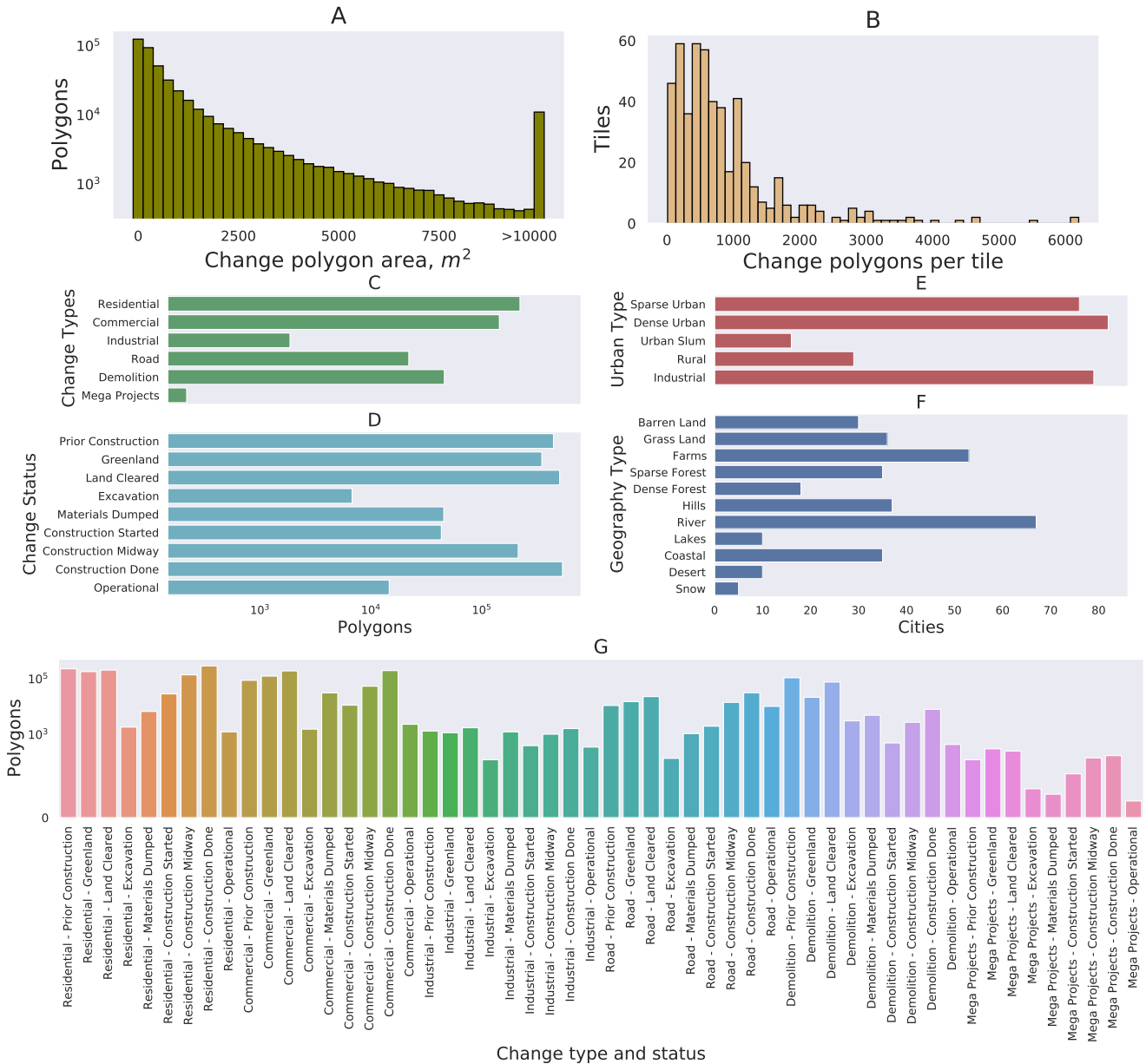


Figure 2. QFabric statistics. Figure A shows number of polygons vs area of polygons, majority of the polygons have small area as most of the polygons represent residential and commercial buildings. Some polygons are exceptionally large belonging to large industrial and mega project classes. Figure B shows distribution of tiles vs polygons per tile. Most of the tiles have less than 1500 polygons. Few tiles have large number of polygons as they belong to fast developing cities where lots of construction has been recorded. Figure C and D shows number of polygons per change type and change status classes, respectively. Figure E and F show number of cities per urban and geography classes, respectively. Figure G show the number of polygons for all combinations of change type - change status.

pixels in area, or 120 pixels across. Figure 2B shows number of tiles vs number of change polygons per tile. This distribution tells us that certain tiles of some cities have fewer number of change polygons, such cities are densely populated and are mostly developed metropolis. Such sparse changes in a densely populated area are particularly difficult to detect using current methods.

For multi-class classification it is important to under-

stand the distribution of classes, specially in the case of imbalanced classes. Figure 2C and 2D show number of samples per class for change type labels and change status labels. In case of change type labels, we can see that we have very few samples for *Mega Projects* class because there are fewer large-scale construction projects across our selected locations when compared to *Residential* and *Commercial* construction projects. This imbalance can be prob-

lematic while training a change type classifier network, but *Mega Projects* have distinct visual cues which should mitigate the relative imbalance and allow network to differentiate them with ease. In Figure 2D, it can be observed that all change type classes have enough samples and the distribution is not imbalanced. This is because our dataset has five dates spread across 5-6 years and therefore we are able to capture all types of change status across multiple dates and tiles. Figure 2G shows distribution of change type and corresponding status. There are few *Mega Projects* that are fully realized within a 5-6 year observation window, and, as such, *Mega Projects* and *Operational* has least number of samples in our dataset.

We also provide geography and urban type statistics for 100 cities in the proposed dataset. Figure 2E shows urban type class distribution. The proposed dataset has fewer cities with urban slums when compared to sparse and dense urban classes. This reflects lower construction activity that is, perhaps, due to the relative lack of growth and high population density in slums. We observe that most of the cities where construction is happening in slums is when large blocks are demolished to build new commercial, residential or industrial complexes. We believe such examples could be quite useful in capturing not only slum-to-urban trends and poverty uplifting events but also pernicious gentrification.

4. Experiments

We split dataset into training, validation and test sets 70 : 20 : 10 by randomly selecting cities. For all our experiments we use RGB bands. Patches of size 512×512 are used by taking a striding window with stride size 512. RGB bands are scaled between 0 – 255 and represented as an 8-bit raster file. For each city grid 6 different masks files are generated. One mask file represents change type label in the change polygons, five other masks files are used to represent the change status of polygons in each date. 90 degree rotations, X and Y flips, imagery zooming of up to 25%, and linear brightness adjustments of up to 50% are applied randomly to training images. We consider four tasks, change detection, change type classification, change status tracking and neighborhood classification.

PyTorch is employed to implement all networks. In order to manage our experiments we use Polyaxon on a Kubernetes cluster and use three computing nodes with eight V100 GPU each.

4.1. Change detection

In change detection task, the goal is to find change polygon from images. We consider two image inputs, bi-date where first date and last date image pairs are input to the network and multi-date where we use all five dates as input to the networks. Based on these input we have two different types of networks, bi-date and multi-date networks.

Bi-date networks are Siamese version of existing fully connected segmentation networks. FC-Siam-Conc [10], FC-Siam-Diff [10], FC-EF[10], BiDate-UNet [23] and BiDate-XDXDUNet are used with date pairs as input. For multi-date input we benchmark using MTLCC [27], LSTM-UNet [23], and MultiDate-XdXdUNet.

Due to the class imbalance of the training data - only $\bar{2}\%$ of the pixels in the training set correspond to buildings. Segmentation models trained with binary cross-entropy (BCE) loss failed to identify building pixels, a problem solved previously for overhead imagery segmentation models [19]. For the semantic segmentation models, we therefore utilized a hybrid loss function that combines the binary cross entropy and intersection over union (IoU) loss with a weight factor α .

$$\mathcal{L} = \alpha \mathcal{L}_{BCE} + (1 - \alpha) \mathcal{L}_{IoU} \quad (1)$$

$\alpha = 0.25$ is used in the loss function to train all change detection networks.

FC-Siam-Conc, FC-Siam-Diff, FC-EF models are trained using Adam optimizer. Initial learning rate is set to $1e - 3$. The model is trained for 150 epochs, and learning rate is decreased by a factor of 0.5 when validation loss failed to improve for 5 epochs. Model training was halted when validation loss failed to improve for 10 epochs.

XdXdUNet is one of the winning solutions from the SpaceNet4 off-nadir challenge. It is a UNet with VGG16 encoder. We modify XdXdUNet by making a Siamese encoder with shared parameters. Similar to XdXdUNet, we also use VGG19 pretrained on ImageNet for the encoder. We fine-tune the BiDate-XdXdUNet on our dataset. Adam optimizer is used with learning rate of $1e - 4$. The model is trained for 150 epochs, and learning rate is decreased by a factor of 0.5 when validation loss failed to improve for 8 epochs. Epoch break off is done when minimum learning rate of $5e - 5$ is reached. Model training is halted when validation loss failed to improve for 10 epochs.

MTLCC [27] uses multi-temporal input to do land-use land-cover pixel-wise classification. Similar to the original network, we give five dates in sequence and reverse to the network as input and train it with the segmentation loss. Input patch size is kept at 512×512 . We use ConvGRU instead of ConvLSTM to allow for larger batch size. We use a batch size of 32. Bidirectional ConvGRU cells size is kept at 256. We use Adam optimizer with learning rate of $1e - 3$ and train the network for 150 epochs. Training was stopped when validation loss failed to improve for 10 epochs. Similar training configuration is used for both LSTM-UNet and MultiDate-XdXdUNet.

Table 2 shows results obtained by different methods for binary change detection task. We observe that MultiDate-XdXdUNet performs best overall and BiDate-XdXdUNet

Network	Dates	Precision	Recall	F1	BA
FC-Siam-Conc [10]	2	33.13	58.73	42.36	72.78
FC-Siam-Diff [10]	2	33.82	59.65	43.17	73.34
BiDate-UNet [23]	2	41.57	58.83	48.72	74.82
FC-EF [10]	2	40.08	66.11	49.90	77.56
BiDate-XdXdUNet	2	46.01	68.44	55.03	79.75
MTLCC [27]	5	52.16	68.03	59.04	80.55
LSTM-UNet [23]	5	55.81	67.90	61.26	80.96
MultiDate-XdXdUNet	5	56.24	69.48	62.16	81.74

Table 2. Binary change detection result using different methods. Metrics used are described in [22].

performs best among all bi-date input networks. We believe MultiDate-XdXdUNet benefits from pretrained weights and multi-date input.

4.2. Change type classification

Network	Dates	mIoU	SeK	Precision	F1	OA
FC-Siam-Conc [10]	2	65.15	26.98	79.19	73.56	71.33
FC-Siam-Diff [10]	2	65.14	27.15	79.24	71.43	71.43
BiDate-UNet [23]	2	65.08	28.03	79.77	74.11	72.01
FC-EF [10]	2	64.98	29.33	80.07	74.76	73.07
BiDate-XdXdUNet	2	68.78	31.26	81.03	76.98	75.34
MTLCC [27]	5	72.63	37.49	83.93	80.99	79.83
LSTM-UNet [23]	5	72.66	38.24	84.30	81.28	80.13
MultiDate-XdXdUNet	5	76.52	36.83	83.57	81.64	80.65

Table 3. Change type classification into 6 classes using different networks.

For change type classification we keep training configurations for all the networks exactly same as binary change detection task. The mask now has 7 labels – 6 change types and 1 for no change. Binary change detection task is now a multi-class classification task.

We report mean intersection over union (mIoU), separated Kappa (SeK), precision, F1-Score and overall accuracy (OA). Separated Kappa alleviates the influence of label imbalance. OA, mIoU, and SeK are defined in [31]. We compute all metrics on 6 change type classes and ignore the no change class. This makes sure that we compare only change type classes as change pixels are very few compared to no change pixels. Other important factor is that binary change detection network is much better at detecting no change vs change than change type networks.

Table 3 shows result for different networks on test set for change type classification task. We observe that LSTM-UNet gives best selected Kappa values (SeK) and precision whereas MultiDate-XdXdUNet is better on mIoU, F1-Score and overall accuracy (OA). Figure 3 shows confusion matrix of MultiDate-XdXdUNet on test set for change type classification task. We observe that the network confuses between *Commercial* and *Industrial* classes. However, this issue may stem from confusion between these two classes for during dataset annotation.

Residential	0.8	0.09	0.01	0	0	0
Commercial	0.1	0.77	0.13	0	0	0
Industrial	0.03	0.11	0.73	0.01	0.01	0.01
Road	0.01	0	0.05	0.9	0.01	0.01
Demolition	0.06	0.01	0.02	0.03	0.96	0
Mega Projects	0	0.02	0.06	0.06	0.02	0.98

Figure 3. Confusion matrix showing results obtained by MultiDate-XdXdUNet on test set for change type detection.

4.3. Change status tracking

Method	mIoU	SeK	Precision	F1	OA
MTLCC [27]	55.94	24.03	63.07	51.03	44.89
LSTM-UNet [23]	55.93	25.36	64.63	52.77	46.35
MultiDate-XdXdUNet	56.36	27.19	67.22	54.49	48.16

Table 4. Change status classification into 9 classes using different networks.

Prior Construction	0.43	0	0	0	0	0.2	0.16	0.13
Greenland	0.02	0.86	0.04	0.01	0	0	0.02	0.03
Land Cleared	0	0.04	0.47	0.41	0.21	0.02	0	0
Excavation	0	0.05	0.43	0.43	0.33	0.03	0	0
Materials Dumped	0.02	0	0.05	0.1	0.4	0.04	0.02	0.02
Construction Started	0.08	0	0	0.05	0.04	0.67	0.08	0.08
Construction Midway	0.13	0.02	0	0	0	0.09	0.33	0.09
Construction Done	0.12	0.01	0	0	0.01	0.1	0.15	0.35
Operational	0.18	0.01	0	0	0	0.05	0.18	0.26

Figure 4. Confusion matrix showing results obtained by MultiDate-XdXdUNet on test set for change status tracking.

Similar to change type classification we keep all training configuration same as in binary change detection task. Since change status classification is a multi-date task, we only train multi-date networks. The decoder part of all three networks are modified to give output for each date. Change status mask for five date is used for training. Mask has 10 labels, 9 change status and 1 for no change. Same metrics as change type classification are reported.

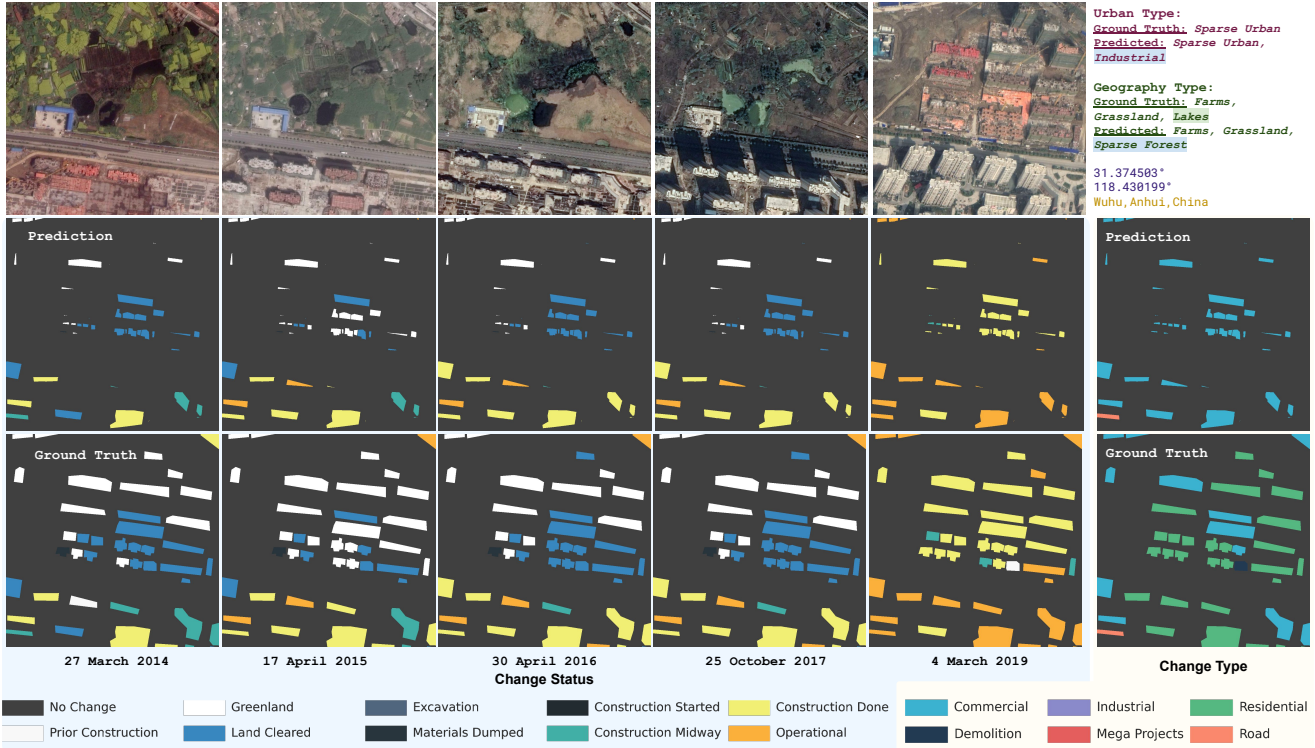


Figure 5. Change type, status and neighborhood urban and geography type classification on a region from one of the city in test set.

Table 4 reports metrics from three networks on test set. We observe that MultiDate-XdXdUNet performs best. We plot confusion matrix for this network as shown in figure 4. We observe two separate groups of classes where network gets confused, first is group of *Prior Construction*, *Construction Midway*, *Construction Done*, and *Operational*, and second is *Land Cleared*, *Excavation*, and *Materials Dumped*.

4.4. Neighborhood Classification

Identifying urban and geographic features around the change polygon is a multi-label image classification problem. We train two ResNet-50 for urban classification and geographic classification tasks. Pretrained weights on ImageNet are used during training. Binary cross entropy is used as a loss for multi-label task. We resize 512×512 input images to 224×224 . Standard hyper-parameters for ResNet50 transfer learning are used. Table 5 reports results for these two tasks. Figure 5 shows visual result for all four tasks on a 1024×1024 random crop from one of the test sample.

5. Conclusions

This paper presents QFabric, an open source change detection dataset, comprising of 450,000 change polygons annotated across 504 locations in 100 different cities covering a wide range of geographies and urban fabrics. Change

Metric	Urban Type	Geography Type
Exact Match Ratio	53.09	47.18
Accuracy	70.69	68.55
Hamming Loss	9.97	6.08
Precision	90.05	93.94
Recall	71.00	68.74
F1-Score	76.87	76.30

Table 5. Different metrics on test set prediction for urban type and geography type multi-label classification using ResNet50.

polygons in QFabric are manually labeled into 6 change type and 9 change status classes, and change polygons neighborhood are labeled into 5 urban type and 11 geography type categories. A number of state-of-the-art deep learning models are employed to perform four major tasks on QFabric, namely change detection, change type classification, change status tracking and neighborhood classification, to serve as baseline for the dataset.

The intent behind creating and presenting QFabric dataset is to encourage development of novel deep learning models for change detection and classification, change status tracking and neighbourhood classification that improve upon our benchmarks. In future, we will be adding more annotated polygons to the dataset, and improve and update labels based on feedback from research community.

References

- [1] Arabi Mohammed El Amin, Qingjie Liu, and Yunhong Wang. Convolutional neural network features based change detection in satellite images. In Xudong Jiang, Guojian Chen, Genci Capi, and Chiharu Ishii, editors, *First International Workshop on Pattern Recognition*, volume 10011, pages 181 – 186. International Society for Optics and Photonics, SPIE, 2016.
- [2] C. Benedek and T. Sziranyi. Change detection in optical aerial images by a multilayer conditional mixed markov model. *IEEE Transactions on Geoscience and Remote Sensing*, pages 3416–3430, 2009.
- [3] H. Butler, S. Gillies, and T. Schaub. The geojson format. RFC 7946, RFC Editor, 08 2016.
- [4] R. Caye Daudt, B. Le Saux, A. Boulch, and Y. Gousseau. Urban change detection for multispectral earth observation using convolutional neural networks. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2018.
- [5] Rodrigo Caye Daudt, Bertrand Le Saux, Alexandre Boulch, and Yann Gousseau. Guided anisotropic diffusion and iterative learning for weakly supervised change detection. In *Computer Vision and Pattern Recognition Workshops*, June 2019.
- [6] Hao Chen and Zhenwei Shi. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. *Remote Sensing*, 12(10), 2020.
- [7] Y. Chen, X. Ouyang, and G. Agam. Mfcnet: End-to-end approach for change detection in images. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 4008–4012, 2018.
- [8] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *CVPR*, 2018.
- [9] Rodrigo Caye Daudt, Bertrand Le Saux, Alexandre Boulch, and Yann Gousseau. Urban change detection for multispectral earth observation using convolutional neural networks. In *IEEE International Geoscience and Remote Sensing Symposium, IGARSS*. IEEE, 2018.
- [10] Rodrigo Caye Daudt, Bertrand Le Saux, and Alexandre Boulch. Fully convolutional siamese networks for change detection. In *ICIP*, 2018.
- [11] Rodrigo Caye Daudt, Bertrand Le Saux, Alexandre Boulch, and Yann Gousseau. Multitask learning for large-scale semantic change detection. *Computer Vision and Image Understanding*, 187:102783, 2019.
- [12] A. M. El Amin, Q. Liu, and Y. Wang. Zoom out cnns features for optical remote sensing change detection. In *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pages 812–817, 2017.
- [13] Adam Van Etten, Dave Lindenbaum, and Todd M. Bacastow. Spacenet: A remote sensing dataset and challenge series. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [14] Adam Van Etten, Jesus Martinez Manso, Daniel Hogan, Ryan Lewis, Jacob Shermeyer, and Christyn Zehnder. SN7: Multi-Temporal Urban Development Challenge. <https://spacenet.ai/sn7-challenge/>, 2020.
- [15] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015.
- [16] Enqiang Guo, Xinsha Fu, Jiawei Zhu, Min Deng, Yu Liu, Qing Zhu, and Haifeng Li. Learning to measure change: Fully convolutional siamese metric networks for scene change detection. *CoRR*, abs/1810.09111, 2018.
- [17] Ritwik Gupta, Bryce Goodman, Nirav Patel, Ricky Hosfelt, Sandra Sajeed, Eric Heim, Jigar Doshi, Keane Lucas, Howie Choset, and Matthew E. Gaston. Creating xbd: A dataset for assessing building damage from satellite imagery. In *CVPR Workshops*, pages 10–17, 2019.
- [18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- [19] Darius Lam, Richard Kuzma, Kevin McGee, Samuel Doolley, Michael Laielli, Matthew Klaric, Yaroslav Bulatov, and Brendan McCord. xvview: Objects in context in overhead imagery, 2018.
- [20] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollr. Microsoft coco: Common objects in context, 2015.
- [21] Lichao Mou, Lorenzo Bruzzone, and Xiao Xiang Zhu. Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery. *CoRR*, abs/1803.02642, 2018.
- [22] Maria Papadomanolaki, Maria Vakalopoulou, and Konstantinos Karantzas. A Deep Multi-Task Learning Framework Coupling Semantic Segmentation and Fully Convolutional LSTM Networks for Urban Change Detection. *IEEE Transactions on Geoscience and Remote Sensing*, Feb. 2021.
- [23] M. Papadomanolaki, S. Verma, M. Vakalopoulou, S. Gupta, and K. Karantzas. Detecting urban changes with recurrent neural networks from multitemporal sentinel-2 data. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 214–217, 2019.
- [24] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. *IEEE Transactions on Image Processing*, 14(3), 2005.
- [25] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.
- [26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015.
- [27] Marc Ruwurm and Marco Krner. Multi-temporal land cover classification with sequential recurrent encoders. *ISPRS International Journal of Geo-Information*, 7(4), 2018.

- [28] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. *BMVC*, pages 61.1–61.12, 2015.
- [29] ASHBINDU SINGH. Review article digital change detection techniques using remotely-sensed data. *International Journal of Remote Sensing*, 10(6):989–1003, 1989.
- [30] M. Vakalopoulou, K. Karatzalos, N. Komodakis, and N. Paragios. Simultaneous registration and change detection in multitemporal, very high resolution remote sensing data. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2015.
- [31] Kunping Yang, Gui-Song Xia, Zicheng Liu, Bo Du, Wen Yang, and Marcello Pelillo. Asymmetric siamese networks for semantic change detection, 2020.
- [32] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4353–4361, 2015.
- [33] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu. Change detection based on deep siamese convolutional network for optical aerial images. *IEEE Geoscience and Remote Sensing Letters*, 14(10):1845–1849, 2017.