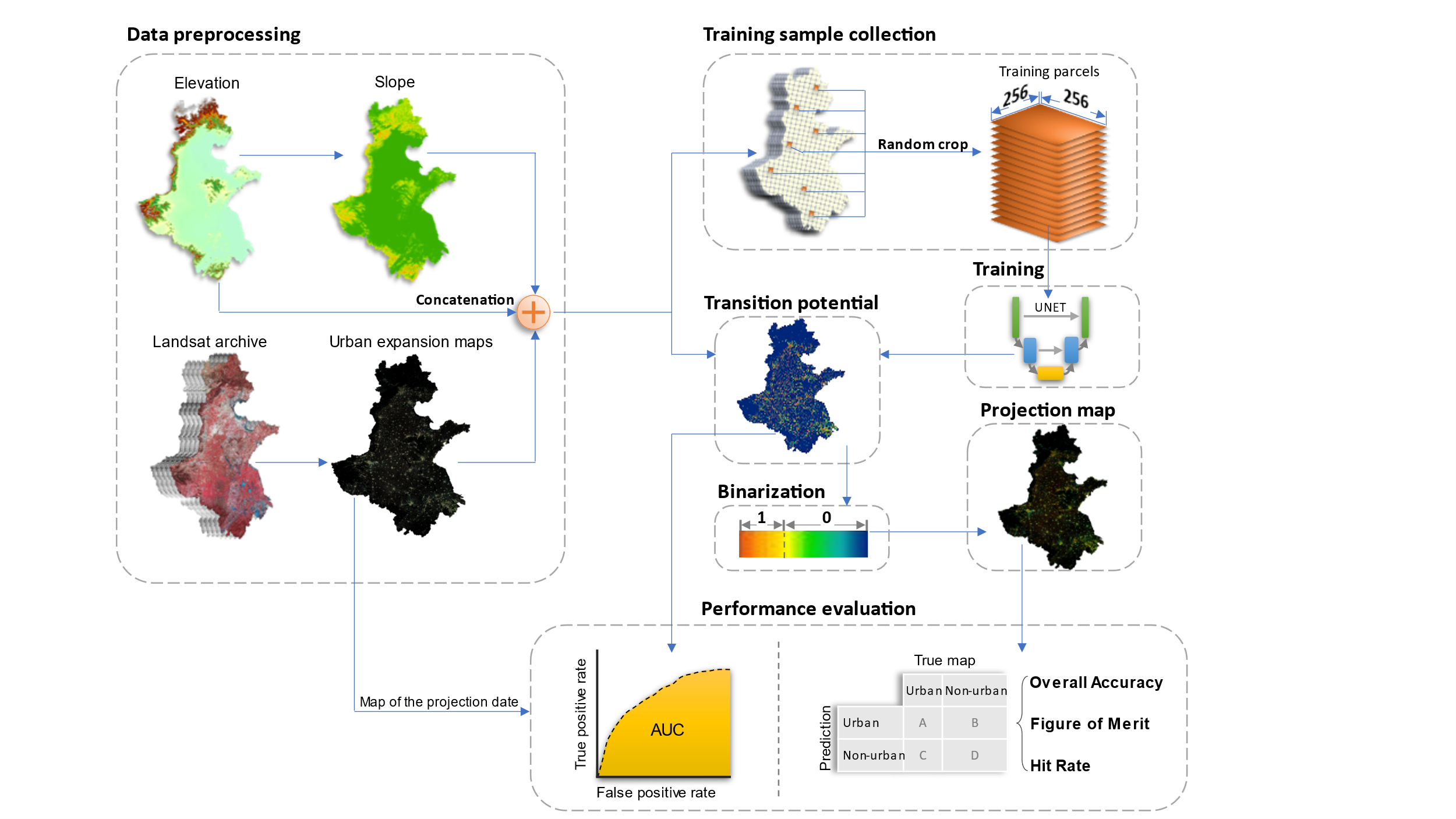
# 2. Data and Method

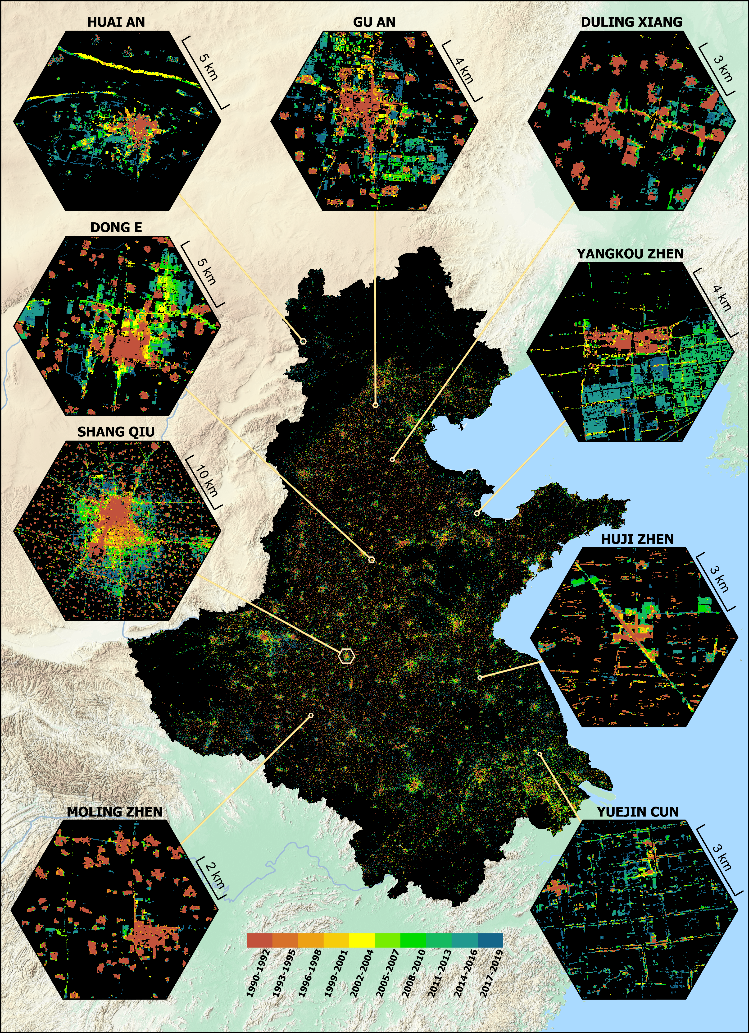
The research workflow is summarized in Fig. The Landsat archive was used to mapping the urban development from 1990 to 2019 with a 3-year interval (paper-1). Digital Elevation Model (DEM) and slope were used to improve the projection performance. All raster data had the same 30m resolution and were concatenated to increase efficiency. The training samples were randomly cropped from the training image (i.e., image with an urban map in earlier date) at the size of 256 × 256 (i.e., the width and heights were 256 pixels). The UNET was trained on these samples and then applied to the base image (i.e., image with a later urban map) to produce the transition potential map (i.e., image where each pixel indicating the potential to be urban lands in the future). Lastly, a threshold was determined to binarize the transition potential map into a projection map where pixels with the value 1 were the projected urban land and 0 the non-urban land.



## 2.1 Data preprocessing

The raster data used in this study were summarized in table. The urban dynamics maps were obtained from (paper-1). This product spanned from 1990 to 2019 with three-year intervals and had a consistently high accuracy of > 94% across years because the use of temporal features successfully delineates urban land from fallow cropland and a further consistent check removed unreasonable urban developments (ref). The digital elevation model (DEM) was obtained from the Shuttle Radar Topography Mission (SRTM) and the slope data was derived from the DEM. The DEM and slope were used to assist the UNET to learn urban development patterns in complicated topology conditions, which have been proven effective in previous studies (ref).

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Source** | **Resolution** | **Time** |
| Urban dynamics | Wang et. al. | 30m | 1990-2019 |
| DEM | SRTM | 30m | 2000 |
| Slope | Derived from DEM | 30m | 2000 |



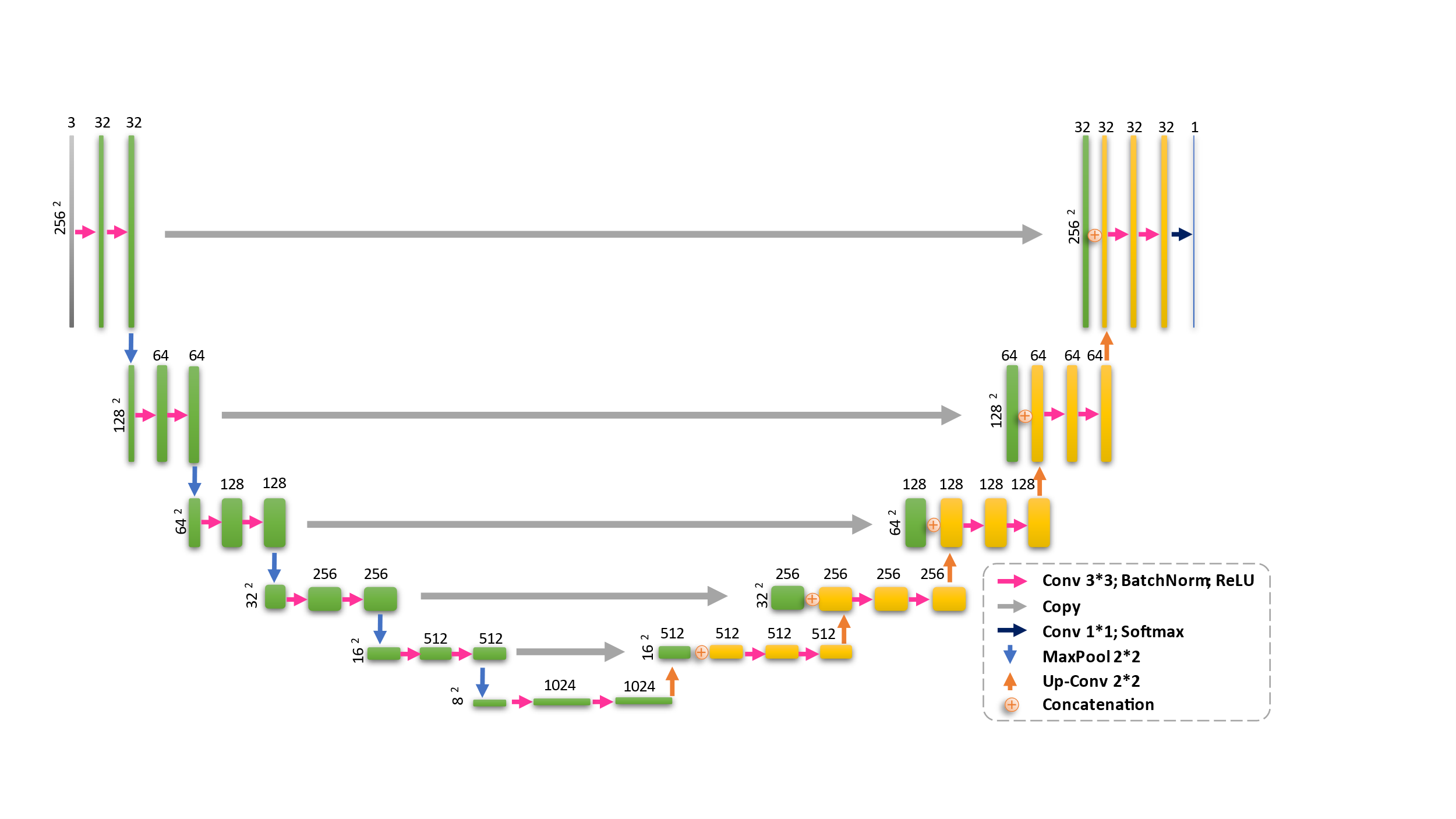
## 2.2 Control sample collection

Control samples were collected at the Google Earth Engine platform for its parallel computation advantages (ref). The *neighborhoodToArray* module was used to randomly crop parcel samples from the training image. We set the neighborhood size to 256 following common data science practices (ref) and collected 25,000 samples, of which 20,000 were used for the model training and the rest 5,000 for validation.

## 2.3 Training the UNET

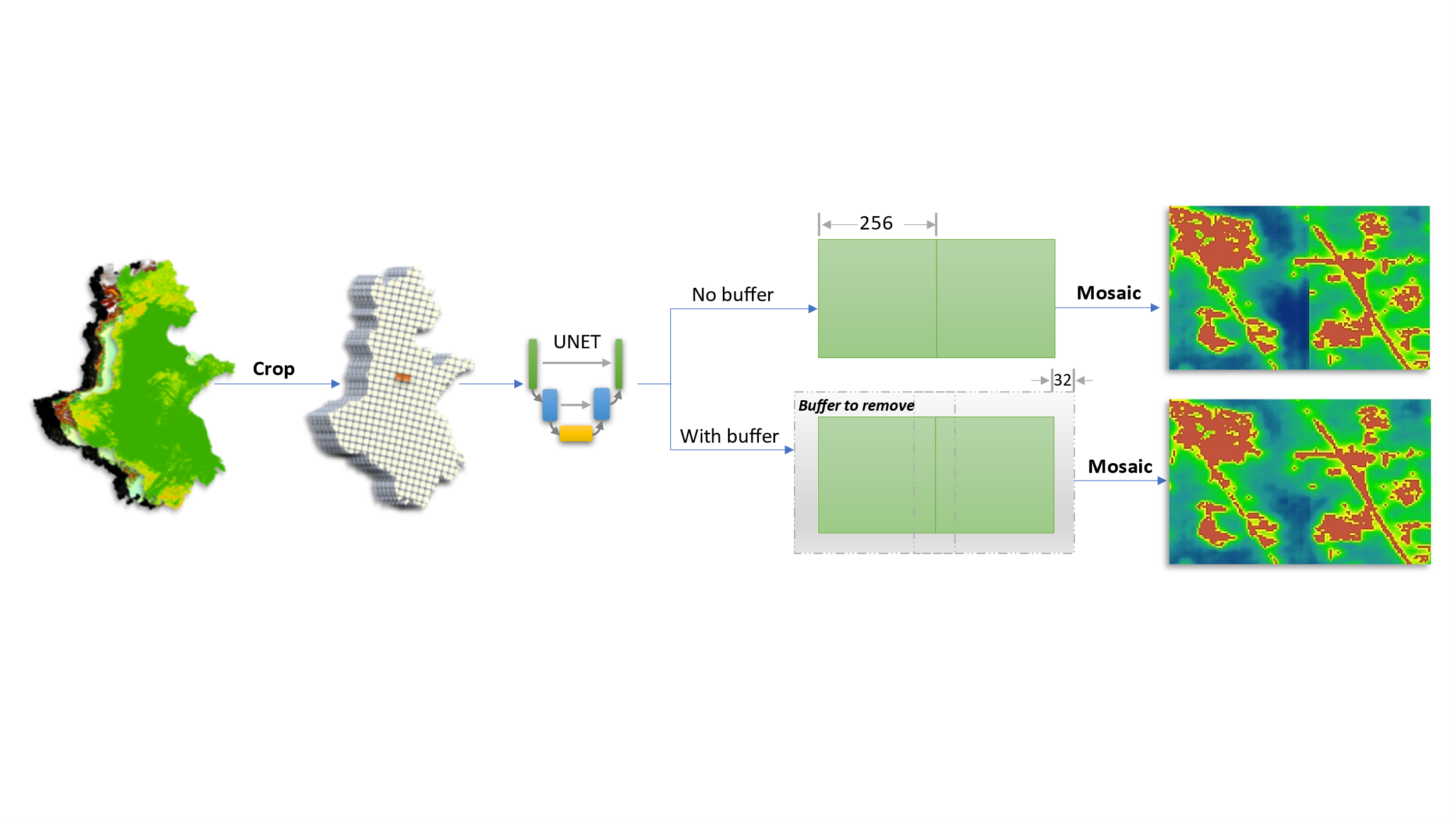
UNET is a deep convolutional network that is commonly used for image segmentation (ref). Down-sampling, up-sampling, and skip-connections are the primary components for UNET (ref). The down-sampling enables UNET to extract the general context of the input image, the up-sampling refined the context pattern to precise shapes, and the skip-connections balance the generalization of down-sampling and the localization of up-sampling (ref). The output of the UNET is an image of the same size as the input data, where each pixel indicating the probability belonging to a preset category.

The UNET was trained on parcel images with the size of 256 × 256 pixels (fig). Each training image parcel includes a 3-layer input data (an early data urban map, a DEM, and a slope map) and a single-layer target data (a later data urban map indicating the true urban expansion). The 3-layer input data, during training, will be transformed to the size of 8 × 8 pixels after 5 down-sampling (i.e., max pooling with kernel size and stride to be two) processes, and then converted to a single-layer output image with the original size of 256 × 256 pixels over another 5 up-sampling (i.e., transpose convolution with kernel size and stride to be two) operations. The mean squared error (MSE) was used to measure the difference between the output and the target image. A total of > 31 million parameters were involved in the training process and the derivative of each parameter to the MSE was calculated via a backpropagation process. The parameters were updated according to the corresponding derivatives by using the Adam algorithm with a learning rate of 0.001, beta-1 as 0.9 and beta-2 as 0.99. We iterated this training process for 200 epochs (an epoch refers to the UNET have seen all 20,000 training samples) because the model will overfit the training samples with more training epochs. The validation samples had the same structure as the training sample but did not update the model parameters and were used to assess the UNET on unseen data.



## 2.4 Producing the transition potential map

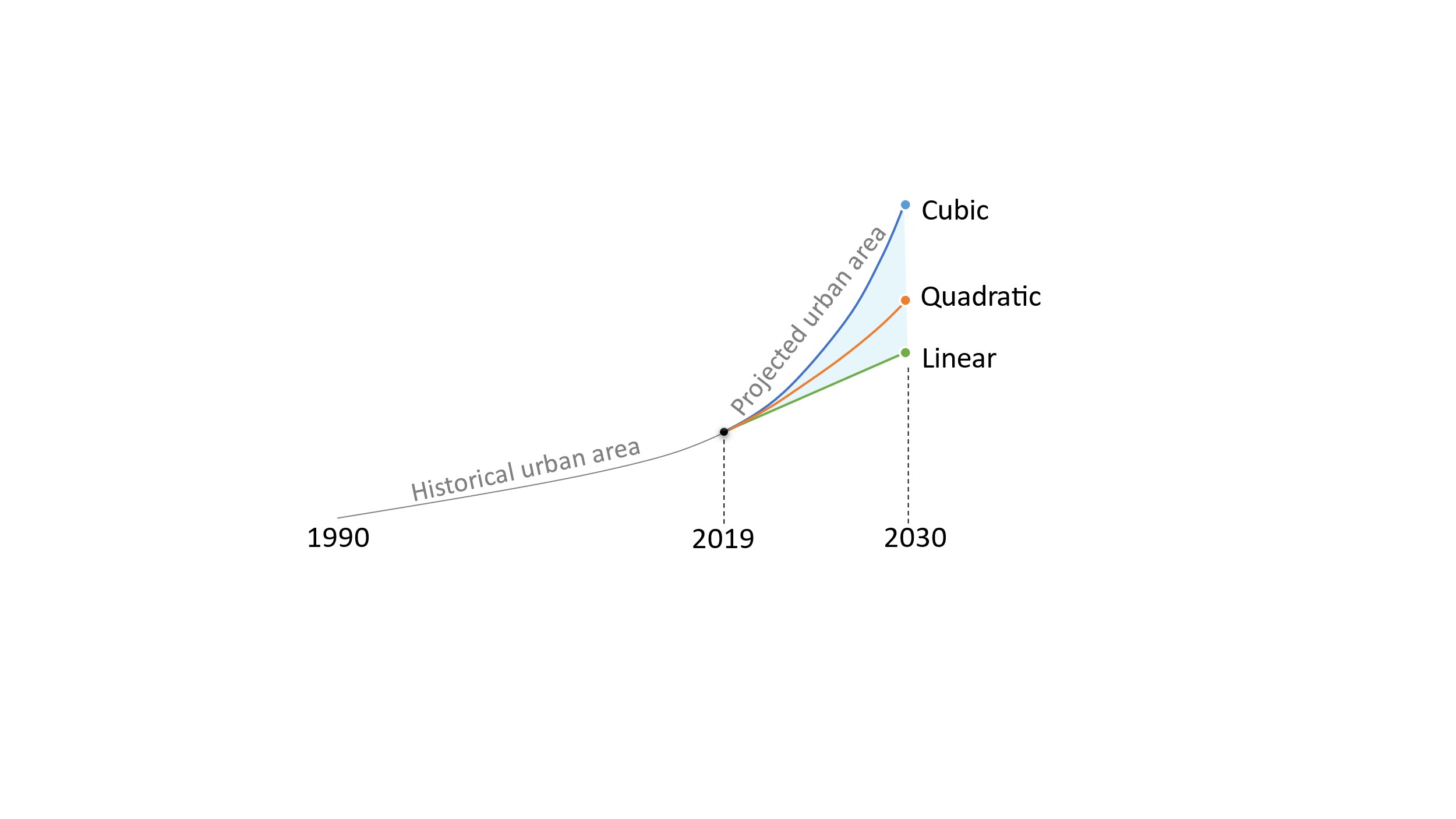
The base image was split into image parcels, then supplied to the trained UNET to produce the separate output, and then mosaiced into the transition potential map. The pixel values of this map ranged from 0 to 1 and indicated the probability to be an urban pixel in the projection date. To reduce the “edge effect”, the split image parcels were buffered with 32 additional pixels at each of the four edges compared to training image parcels, and the buffer was removed before been mosaiced into the final transition potential map. Note although the UNET was trained on image parcels of 256 × 256 pixels, it can be applied to the buffered image parcels of 320 × 320 pixels (i.e., the original size of 256 plus two buffer zones of size 32) (ref).



## 2.5 Producing the projection map

The projection map was created by binarizing the transition potential map. We computed the pixel count of the transition potential map at an interval of 0.0001, then accumulated these pixel counts from the highest value to the lowest, then selected the accumulated value that was the closest to the urban pixel count in the projection year, and used it as the threshold to binarize the transition potential map.

The UNET was validated first and then used to project urban map in 2030. The validation took the urban maps of 1993-1995 and 2005-2007 to train a UNET model, then it was applied to the urban map of 2005-2007 to project to 2017-2019. The true urban map of 2017-2019 was used to assess the performance of the projection. Similarly, we used the urban map of 2005-2007 and 2017-2019 to train other models to project urban map of 2030. The urban area of 2030 was determined by regressions on the historical urban area: a linear, a quadratic, and a cubic fitting were applied on the past urban areas to determine the urban area in 2030. These urban areas were then used as thresholds to binarize the transition potential map and produced three potential urban maps in 2030.



## 2.6 Assessing the performance of the projection

The transition potential map was assessed using the area under the curve (AUC) of the receiver operating characteristic (ROC). The projected map was evaluated via map-overlay metrics and landscape metrics. In this study, as the threshold to binarize the transition potential map increased from 0 to 1, the portion of pixels simulated as urban in the projection map but were non-urban in the actual map (e.g., false-positive rate) and the portion of correctly simulated urban pixels (e.g., true positive rate) also changed. By plotting all false-positive/true-positive values in an x-y plane, the ROC (the line that connects all points from left to right) and the AUC (the region between ROC and the x-axis) can be computed. The AUC ranges from 0.5 to 1, indicating models of random to perfect, have proven to be effective to evaluate the transition potential map (ref).

The map-overlay metrics selected in this study were overall accuracy (OA), the hit rate, and the figure of merit (FoM). The OA reflect the model performance on both urban and non-urban predictions but was prone to saturate by non-urban pixels that were more easily to be simulated and taking up most of the pixels during a prediction (ref). The hit rate showed the model performance on predicting urban pixels evaluated from the true map, and the FoM reflects that from both the predicted and the true map (ref). The map-overlay metrics were calculated as below:

|  |  |  |
| --- | --- | --- |
|  | OA = (A + D) / (A + B + C +D) | (1) |
|  | FoM = A / (A + B + C) | (2) |
|  | Hit rate = A / (A + C) | (3) |

where A refers to correctly predicted urban pixels (hit), the B refers to the incorrectly predicted urban pixels (false alarm) , C is the incorrectly predicted non-urban pixels (miss), and D is the correctly predicted non-urban pixels (correct rejection).

We used the path number (PN), shape index (SI) and the fractal dimension index (FDI) to evaluate the similarity/difference between the true urban map and the simulated urban map. A patch is a group of urban pixels that were connected to each other in a 3 × 3 neighborhood. The SI measure the shape complexity of a patch, the SI of a square is 1 and it will increase as the patch shape becomes more irregular (ref). the FDI reflects the complexity of patches that varies a greater range of sizes and provides a stable measurement to straight patches (ref). SI and FDI are calculated as follows:

|  |  |  |
| --- | --- | --- |
|  |  | (2) |
|  |  | (3) |

Where is the perimeter of the patch *i*, and the is the area (hecter) of the patch *i*.