International Symposium on Applied Geoinformatics (ISAG-2019)

DIGITAL TERRAIN MODEL GENERATION FROM HIGH RESOLUTION DIGITAL SURFACE MODEL BY USING CONDITIONAL ADVERSARIAL NETWORKS

Alper Çınar [1], Yasin Koçan [2]

[1]Independent Researcher, Ankara, Turkey (alper@r3maps.com);
[2] Middle East Technical University, Geodetic and Geographic Information Technologies, 06800, Çankaya, Ankara (e162499@metu.edu.tr);

*Corresponding Author, Received: 00/XX/201X, Accepted: 00/XX/201X

ABSTRACT: A Digital Terrain Model (DTM) is a representation of the bare-earth with elevations at regularly spaced intervals. This data is captured via aerial imagery or airborne laser scanning. Prior to use, all the above-ground natural (trees, bushes, etc.) and man-made (houses, cars, etc.) structures needed to be identified and removed so that surface of the earth can be interpolated from the remaining points. Elevation data that includes above-ground objects is called as Digital Surface Model (DSM). DTM is mostly generated by cleaning the objects from DSM with the help of a human operator. Automating this workflow is an opportunity for reducing manual work and it is aimed to solve this problem by using conditional adversarial networks. In theory, having enough raw and cleaned (DSM & DTM) data pairs will be a good input for a machine learning system that translates this raw (DSM) data to cleaned one (DTM). Recent progress in topics like 'Image-to-Image Translation with Conditional Adversarial Networks' makes a solution possible for this problem. In this study, a specific conditional adversarial network implementation “pix2pix” is adapted to this domain.

Data for "elevations at regularly spaced intervals" is similar to an image data, both can be represented as two dimensional arrays (or in other words matrices). Every elevation point map to an exact image pixel and even with a 1-millimeter precision in z-axis, any real-world elevation value can be safely stored in a data cell that holds 24-bit RGB pixel data. This makes total pixel count of image equals to total count of elevation points in elevation data. Thus, elevation data for large areas results in sub-optimal input for "pix2pix" and requires a tiling. Consequently, the challenge becomes "finding most appropriate image representation of elevation data to feed into pix2pix" training cycle. This involves iterating over "elevation-to-pixel-value-mapping functions" and dividing elevation data into sub regions for better performing images in pix2pix.

Keywords: *Digital Terrain Model, Pix2pix, Conditional Adversarial Networks, Digital Surface Model, Object Removal*

1. INTRODUCTION

Digital surface and digital terrain models obtained from aerial techniques are very widespread and the results are used in various disciplines such as cartography, natural resources, city planning and water resources analysis. Most applications in these domains require terrain information, however; digital surface models from aerial outputs include all the above-ground natural (trees, bushes etc.) and human-made (houses, towers etc.) structures. To extract the terrain model (DTM) from DSM, there are different methods on the market; nonetheless, none of these methods work perfect in all cases.

Due to need of the bare earth model for these disciplines, DTM extraction is a hot topic. There are some approaches that focus on detecting above-ground objects, eliminating them and interpolating the remaining surface or points to obtain the terrain model. On the other hand, there are different methods based on finding the ground points to interpolate the remaining surface. These methods mostly depend on rule-based techniques that uses slope-based methods, morphological approaches (Weidner & Förstner, 1995), multi-directional scanlines (Perko, Raggam, Gutjahr, & Schardt, 2015), the cloth simulation (Zhang, et al., 2016) and the sparsity driven algorithms (Nar, Yilmaz, & Camps-Valls, 2018). Even there is a big progress achieved in automated DTM extraction methods, there are still some misclassifications in features such as cliffs, hills or small rocks.

On the other hand, there are some software on the market where a human operator can manually edit the elevation model. However, there is a huge effort is necessary to obtain DTM from DSM data.

The DTM extraction methods that discussed above works on different input types such as LIDAR, photogrammetric point clouds or DSM. In some cases, there is only DSM provided without point cloud and it makes DTM extraction more challenging since point clouds contains more information about the terrain.

In our proposed method, deep learning-based approach is used to solve the problem that rule-based techniques cannot do perfectly in all cases. In this method, neural network model learns the differences from the DSM-DTM pairs for different regions and do the extraction task for the regions to predict the terrain model.

2. DATA

1. SATELLITE IMAGERY

1-m spatial resolution DSM is obtained from tri-stereo Pleiades satellite imagery that cover the Iran-Mashhad region. In addition, 1-m spatial resolution orthophoto is used to help the manual DTM extraction from the DSM. The training area contains different regions to test such as steep regions, dense urban area, relatively relief regions.

2. DTM PREPARATION

There are 1376 tiles in total that consist of 256x256 pixels area. For the corresponding area, DTM data is generated by manual editing from DSM in PCI Geomatica Focus software as a ground-truth data. Manual editing takes around 20 hours to obtain high-quality DTM. In addition to DSM data, to detect the above-ground objects more clearly, a high resolution orthophoto is used. For the above-ground objects, polygons are drawn manually and these regions are interpolated with different methods depending on the region or the object type. There are some regions that is not easy to edit manually such as viaducts, road edges, buildings on steep areas.

There are various methods to obtain the DSM that have different advantages and disadvantages. Depending on the DSM extraction process, the shady areas can be either pits or bumps. Depending on the sensing time and the method used, this issue could be a crucial factor that effects the manual editing time for the DTM extraction.

3. METHOD

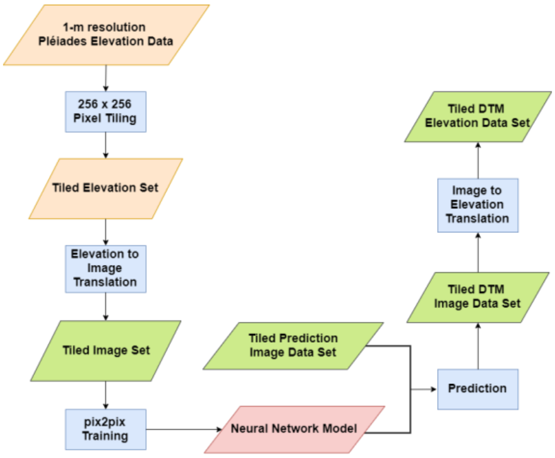

Figure.1 Flow Chart

Our research starts with manually editing Pleiades Elevation data to obtain DTM version of the mentioned elevation data. After generating the manually edited elevation data, both DSM and target DTM data are divided into 256x256 sized tiles to feed them into pix2pix. Prior to use these elevation data in pix2pix, they are transformed to image pairs by using our translation function which is explained in section 3.2. With these image pairs, a neural network model is generated with a nearly 3-day-pix2pix-training session. After training the model, test data pairs are fed into pix2pix model for prediction. Predicted output is then compared with manually edited DTM which is assumed as the ground truth. Every pixel in predicted DTM and ground truth is compared, and the pixel-match ratio is inspected in this paper's quantitative result section 4.1.

3.1. PIX2PIX

Pix2pix is a general-purpose solution for image-to-image translation problems. It uses conditional adversarial networks and effectively works for synthesizing photos from label maps, reconstructing objects from edge maps, among other tasks. Wide usage of pix2pix (Figure 2) showed that it is no longer necessary to hand-tweak mapping functions for image translation tasks. In this research pix2pix is used as it is and only its training data is modified for optimal result.

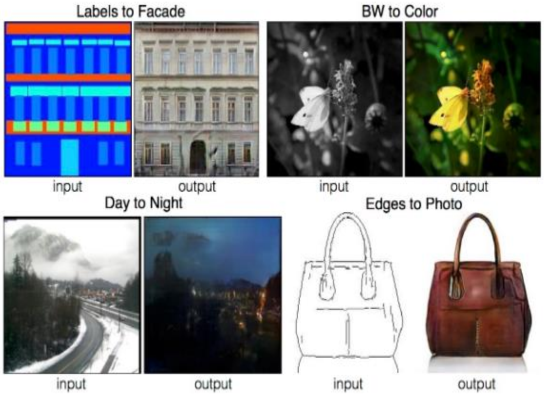

Figure 2. Different uses of pix2pix

Pix2pix is trained with image pairs and output of this training is a neural network model. This model is able to generate new images by looking at the past translations occurred in trained image pairs. Theoretically, having lots of different translation patterns in training set results in a more capable neural network model. In our research, training data pairs are DSMs and DTMs respectively and as distinct DSM-DTM pairs as possible are trained for generating better performing pix2pix model. It can be simply said that the more distinctive training cases, the more capable neural network model gets.

3.2. TRANSLATION FORMULAS

$$S_{elevation}=[s_{1,1} \cdots s_{1,256} \vdots \ddots \vdots s_{256,1} \cdots s_{256,256}] \quad (1)$$

$$T_{elevation}=[t_{1,1} \cdots t_{1,256} \vdots \ddots \vdots t_{256,1} \cdots t_{256,256}] \quad (2)$$

$$f(x)=floor\left(\frac{(x\text{-}Pair_{min})*(2^{(bitdepth)}\text{-}1)}{Pair_{\Delta}}\right) \quad (3)$$

$$S_{image}=\left[f(s_{1,1}) \cdots f(s_{1,256}) \vdots \ddots \vdots f(s_{256,1}) \cdots f(s_{256,256}) \right] \quad (4)$$

$$T_{image}=\left[f(t_{1,1}) \cdots f(t_{1,256}) \vdots \ddots \vdots f(t_{256,1}) \cdots f(t_{256,256}) \right] \quad (5)$$

$$S_{min}= min(S) \quad (6)$$

$$T_{min}= min(T) \quad (7)$$

$$S_{max}= max(S) \quad (8)$$

$$T_{max}= max(T) \quad (9)$$

$$Pair_{min}=min(T_{min},S_{min}) \quad (10)$$

$$Pair_{max}=min(T_{max},S_{max}) \quad (11)$$

$$Pair_{\Delta}=Pair_{max}\text{-}Pair_{min} \quad (12)$$

In order to use pix2pix for elevation data processing, that data should be represented as an image. There are infinite ways to show elevation data as an image but this image representation should preserve the height characteristics and should reveal the smallest changes in elevation data as much as possible. Both elevation data and images have similar data structure, they both can be expressed as 2-dimensional number arrays which means that every elevation point corresponds to an image pixel but there are some limitations in storing image data. Images are consisting of pixels and each pixel has finite amount of data to be stored in it. In our research, elevation data is transformed to an 8-bit grayscale image which means that elevation data must be fit into an 8-bit depth data store.

Our elevation data is expressed in centimeter precision height values where value 0 refers to mean sea level. In order to map every elevation point into these 8-bit storage effectively, a linear value distribution function is developed. Minimum elevation value in DSM “Eq. (1)”, DTM “Eq. (7)” matrices and maximum elevation value in DSM “Eq. (8)”, DTM “Eq. (9)” matrices are found first. With these minimum and maximum values, difference “Eq. (12)” between lowest “Eq. (10)” and highest “Eq. (11)” points in DSM-DTM pair is found. Then, each elevation point in DSM “Eq. (1)” and DTM “Eq. (2)” are transformed to a pixel value in DSM “Eq. (4)” and DTM “Eq. (5)” image matrices by using our translation function “Eq. (3)”.

3.3 TRAINING & PREDICTION

Training and prediction are executed on an average laptop that has an early 2012 CPU, Intel-i7 2860 QM with a 4GB of memory. There were 1367 image pairs and these image pairs trained for 200 generations (epoch) which took 68 hours to complete on this setup. Pix2pix training can be completed in a much shorter time period though. Ideally, executing the training session on a CUDA compatible GPU would generate the model in a much shorter time.

4. EXPERIMENTAL RESULTS

4.1. QUANTITATIVE

Assuming manually edited data as ground truth, it is possible to measure total pixel match-mismatch ratio. When compared with a pixel matching software that implements pixel and intensity slope detector (Vysniauskas, 2009), 706 of 1367 predicted images are exactly matched with the ground truth. 535 of remaining 661 image has small (<2%) differences with the ground truth data. 61 of 1367 image is completely different than ground truth due to artifacts generated by pix2pix.

Labels	Percentage
Exact Match	54.2
Almost Matched	41.1
Mismatch	4.7

4.2. QUALITATIVE

Results that are labeled as “almost matched” have some interesting cases though. There are some cases where pix2pix do the task “better” than the human operator.

4.2.1. Single trees

There are some areas where we miss out in manual DSM data editing. Unlike forests or dense tree areas, single trees are easy to miss in manual editing. There are a couple of unremoved “single” trees in our training data. Despite this, pix2pix model managed to remove these trees (Figure 3.) due to the fact that model seen enough similar cases on training data. We are able to pick 9 different cases where pix2pix detected and removed a single tree where our human editing effort failed to remove those trees from DTM data.

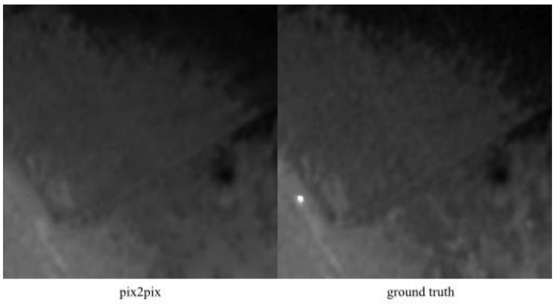

Figure 3. Single tree removal

4.2.2. Hangar

Human editing is also prone to precision errors. In one of the cases, a hangar is recklessly removed from terrain by including its surroundings. When we analyze the case in detail, it is seen that hangar has 2 entry points on its corners with lower elevation values and these entry points are part of the terrain. Pix2pix kept the entry points as a part of terrain model and removed only the building part of hangar (Figure.4).

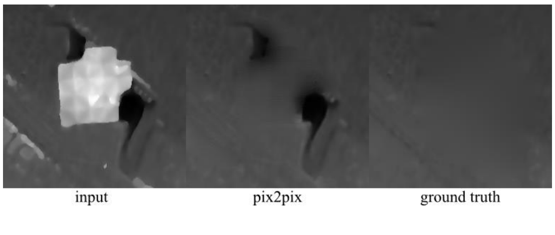

Figure 4. pix2pix corrected human operator

4.2.3. Jitter

Not all differences between ground truth and predictions are in favor of pix2pix, there are some cases where pix2pix generated unwanted results. Pix2pix generates noise on some predictions, this noise is visualized better in a 3D environment (Figure.5). A denoise (blur) filter can be applied to whole elevation area after prediction to reduce this jitter, but this is out of scope for this research since we are interested in capabilities of pix2pix for elevation data.

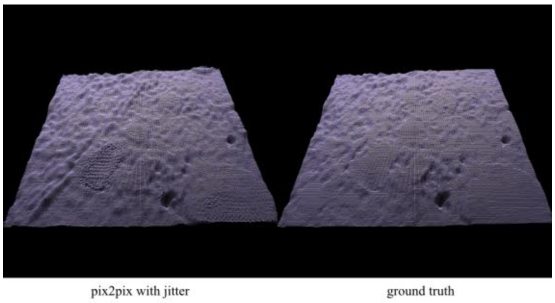

Figure 5. pix2pix jitter problem

4.2.4. Artifacts

Some of the predictions (5%) are different than ground truth due to artifacts generated by pix2pix. This prediction is already marked as “mismatch” by a pixel comparison algorithm in previous section 4.1. When it is analyzed in detail, these artifacts are occurred persistently in same spot of images (Figure.6).

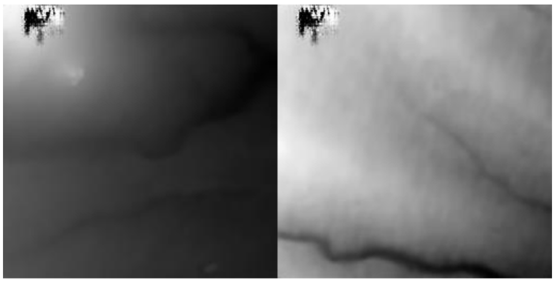

Figure 6. Artifacts

5. CONCLUSION

In this research, pix2pix is evaluated as a DSM to DTM conversion tool and it gave promising results even with minimal effort done to tweak pix2pix for this domain. There are still room for further studies to improve the accuracy of this research.

1. Using a larger and more distinct data set to train.
2. Non-linear value distribution in elevation to image formulas.
3. Working with finer resolution elevation data.
4. Using different color spaces other than grayscale.

ACKNOWLEDGEMENTS

Thanks Airbus Defence and Space for providing sample Pleiades Elevation1 – Mashhad, Iran data.

REFERENCES

Nar, F., Yilmaz, E., & Camps-Valls, G. (2018). Sparsity-Driven Digital Terrain Model Extraction. IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. doi: 10.1109/igarss.2018.8517569

Arefi, H.; Reinartz, P. Building reconstruction from Worldview DEM using image information. SMPR2011, 2011.

Perko, R.; Raggam, H.; Gutjahr, K.; Schardt, M. Advanced DTM Generation from Very High Resolution Satellite Stereo Images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 2015, *II-3/W4*.

Weidner, U.; Förstner, W. Towards automatic building extraction from high resolution digital elevation models. *ISPRS* 1995, *50 (4)*, 38–49.

Vysniauskas, Vytautas. (2009). Anti-aliased Pixel and Intensity Slope Detector. ELECTRONCS AND ELECTRICAL ENGINEERING.

Zhang, W., Qi, J., Wan, P., Wang, H., Xie, D., Wang, X., & Yan, G. (2016). An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. Remote Sensing, 8(6), 501. doi: 10.3390/rs8060501

Kotsarenko, Y., & Ramos, F. (2010). Measuring perceived color difference using YIQ NTSC transmission color space in mobile applications.