Geocentric Pose Analysis of Satellite Imagery Using Deep Learning

Christopher Sun, Jai Sharma, Milind Maiti

March 2022

Abstract

Roughly 6,800 natural disasters occur worldwide annually, and this alarming number continues to grow due to the effects of climate change. Effective methods to improve natural disaster response include performing change detection, map alignment, and vision-aided navigation to allow for the time-efficient delivery of life-saving aid. Current software functions optimally only on nadir images taken ninety degrees above ground level. The inability to generalize to oblique images increases the need to compute an image’s geocentric pose, which is its spatial orientation with respect to gravity. This Deep Learning investigation presents three convolutional models to predict geocentric pose using 5,923 nadir and oblique RGB satellite images of cities worldwide. The first model is an autoencoder that condenses the $256 \times 256 \times 3$ images to $32 \times 32 \times 16$ latent space representations, demonstrating the ability to learn useful features from the data. The second model is a U-Net Fully Convolutional Network with skip connections used to predict each image’s corresponding pixel-level elevation mask. This model achieves a median absolute deviation of 0.335 meters and an $R^2$ of 0.865 on test data. Afterward, the elevation masks are concatenated with the RGB images to form $256 \times 256 \times 4$ inputs of the third model, which predicts each image’s rotation angle and scale, the components of its geocentric pose. This Deep Convolutional Neural Network achieves an $R^2$ of 0.904 on test data, significantly outperforming previous models designed by researchers. The high-accuracy software built in this study contributes to crucial procedures that can accelerate disaster relief and save human lives.
Contents

1 Introduction .................................................. 3

2 Methods and Results ........................................... 3
   2.1 Data Used For Experimentation .............................. 3
   2.2 Outlier Removal ............................................ 4
   2.3 Autoencoder Model ......................................... 4
      2.3.1 Visualization of Latent Space Representations ....... 5
      2.3.2 Autoencoder Model Performance ...................... 5
   2.4 U-Net Elevation Model ..................................... 6
      2.4.1 U-Net Elevation Model Performance ................... 6
      2.4.2 Interpolation of Invalid Data ......................... 7
      2.4.3 Visualizations ......................................... 7
   2.5 Geocentric Pose Model ..................................... 8
      2.5.1 Geocentric Pose Model Performance .................. 8
   2.6 Geocentric Pose Ensemble Model Metrics Summary ........ 9

3 Discussion .................................................... 10
   3.1 Insights from Deep Learning Models ...................... 10
   3.2 Comparison to Prior Research ............................... 10

4 Conclusion ................................................... 11
1 Introduction

With the number of natural disasters on the rise, there is an urgent need for land management technologies that provide disaster relief. For unmanned aerial vehicles (UAVs) to effectively respond to natural disasters, they must have adaptable and versatile software that perform a wide array of necessary functionalities, such as detecting objects, monitoring changes on the land below, and mapping surroundings to allow for communication with other UAVs and humans. While researchers like Mishra et al. [1] have implemented object detection algorithms to potentially assist in search and rescue operations, there has been less progress in the tasks of vision-aided navigation and map alignment for disaster relief. The core technology behind these tasks is finding the geocentric pose of an aerial vehicle. The geocentric pose consists of three components: the relative elevation above the ground, the angle of orientation with respect to gravity, and the scale between the actual and apparent sizes of objects on the ground. Currently, geocentric pose is easiest to predict when UAVs take near-nadir images, which are images taken around ninety degrees and almost directly above the ground. The geocentric pose of oblique, non-nadir images is harder to predict, creating the need for deep learning models to provide a reliable and accurate solution.

Objective  Our research objective was to extract features from near-nadir and oblique RGB satellite images to predict each image’s geocentric pose, using an ensemble model methodology involving two convolutional neural networks.

![Ensemble model to predict geocentric pose.](image)

2 Methods and Results

This paper has combined the Method and Results sections into one section. After each model is discussed, the results of experimentation for that model will be reported.

2.1 Data Used For Experimentation

The data used for this research was the Urban Semantic 3D Data Set, publicly available from the IEEE DataPort [2]. The data set contained 5,923 RGB images, their pixel-wise elevations, and their measurements of scale and angle. The images were taken from four locations in North and South America.

| Location          | No. of Images |
|-------------------|---------------|
| San Fernando, AR  | 2325          |
| Atlanta, USA      | 704           |
| Jacksonville, USA | 1098          |
| Omaha, USA        | 1796          |

Elevation was given in centimeters, scale was given in pixels per centimeter, and angle was given in radians. Due to computational constraints, we had resized the images from $2048 \times 2048 \times 3$ pixels to $256 \times 256 \times 3$.  

3
2.2 Outlier Removal

The data set contained elevation masks with invalid pixel values that were either improbably large or NaN. For reference, the tallest building in Jacksonville is 171 feet, which is roughly 5,212 centimeters. However, some elevation masks of Atlanta images in the data set contained pixel values greater than 5,212, demonstrating the invalidity of the outliers. The solution to this problem was Location-based Thresholding, which involved finding pixel value thresholds for each city included in the data set. We found that images from Jacksonville and Omaha did not contain any outliers. For Atlanta images, we counted the number of pixels that exceeded 4,000 centimeters for each image. If this number was greater than 100, we determined that the image indeed contained a building taller than 4,000 centimeters, but if this number was less than 100, we interpolated all said pixels with the median elevation in the image. For San Fernando images, the outlier threshold was a hard cap at 3,000 centimeters, because the landscape was fairly even and did not contain tall buildings. These thresholds were all experimentally determined since there was no objective way to best deal with the outliers in the data set. In particular, the challenge was finding which large pixel values were valid, because of the presence of a tall building, and which were invalid. Future research should explore optimal outlier removal methods for this data set.

2.3 Autoencoder Model

Before embarking on the ensemble model framework, we trained an autoencoder on the RGB images to gauge the practicality of the deep learning approach. Our reasoning was that if meaningful latent space representations could be extracted from the RGB images, the other convolutional models would be able to extract important features to predict the geocentric pose.

![Autoencoder Model](image)

The encoder used convolutional operations to condense the input images to matrices of shape $32 \times 32 \times 16$. These intermediate matrices were then flattened to a shape of $16384 \times 1$, after which a Dense Neural Network was used to arrive at the latent space representation of shape $512 \times 1$. Finally, Multidimensional Scaling was used to visualize these high dimensional latent space representations on a 2D plane, as shown below.

Figure 2: Autoencoder Model trained with 16,823,107 parameters. The autoencoder consisted of an encoder (left) and a decoder (right), and the goal was to make the decoded images as similar to the original input images as possible. Here, a bottleneck architecture approach is used for this task.
2.3.1 Visualization of Latent Space Representations

Figure 3: Multidimensional Scaling (MDS) used to visualize the latent spaces representations of the encoded images. Each point represents one encoded image, and the images are colored by magnitude of scale and angle, respectively. It must be noted that these visualizations do not capture all nuances of the high dimensional data, so these visualizations are inaccurate to an extent.

An inverse architecture of the encoder was used to decode the latent space representations back into images of shape $256 \times 256 \times 3$. The autoencoder, made by combining the encoder and decoder into one function, was trained with the goal of retrieving decoded images which were as similar to the original input images as possible.

**Results for Scale Visualization** No clear association can bee seen from the scale encoding visualizations, implying that the trained autoencoder may not have learned effective features for the image scale.

**Results for Angle Visualization** Contrary to the results of the scale encoding visualizations, the encoding visualizations for the angle show a clearer association. Specifically, images with higher angles are aggregated to the left while images with lower angles are aggregated to the right. There are two main significances of this result. First, this visualization demonstrates that there is a relationship between an image and its angle. Second, the utilized investigation workflow reveals that this relationship is learnable by a convolutional model. In other words, it is feasible to pursue the U-Net model and Ensemble model and predict the Geocentric Pose of a UAV given an RGB image because it is possible to extract useful features to learn this relationship.

2.3.2 Autoencoder Model Performance

- Find MAE

![Original Images](image1.png) ![Autoencoded Images](image2.png)

Figure 4: The original images are compared to the resulting images after passing through the autoencoder.

The similarities between the original images and the autoencoded images shows that the model was able to extract important features to reconstruct the images. This result validated the proposed Deep Learning methodology. Accordingly, we proceeded to build the Geocentric Pose Ensemble Model.
2.4 U-Net Elevation Model

We trained a fully convolutional neural network to predict the elevation mask of each RGB image. The model was based on the U-Net architecture designed for a wildfire classification task, but included modifications such as having more convolutional blocks. The filter size was 5x5 throughout the model. The first convolution involved 32 filters, and this number was doubled for each layer during downsampling and halved for each layer during upsampling. Batch Normalization and ReLU activation were used throughout the model. The model was configured with an 85-15% train-validation split. The model was trained for around 700 epochs, until the validation loss plateaued, using Adam optimization, a learning rate of 0.001, and a mini batch size of 32 images. Images with NaN values were excluded from training so that the model was trained only on valid data.

The U-Net Elevation Model was used to predict each RGB image’s elevation mask containing pixel-wise heights in px/cm (pixels per centimeter).

2.4.1 U-Net Elevation Model Performance

The learning curves above show that model still has potential to learn and has not fully converged, since the validation loss still seems to be trending downwards after more than 700 epochs. However, because of a widening gap between

![Figure 5: The U-Net Elevation Model contained 23,966,337 parameters.](image)

![Figure 6: text here](image)
the train and validation loss, as shown by the figure, we decided to stop training the model. The model achieved an $R^2$ metric of 0.926 on the train set and 0.865 on the validation set.

### 2.4.2 Interpolation of Invalid Data

Following training, the NaN values in the images were replaced by the exact elevation model predictions. This served as a data cleansing and quality control mechanism that allowed all 5,923 images to be used for the Geocentric Pose Model.

### 2.4.3 Visualizations

Figure 7: The elevation masks predicted by the U-Net model are compared to the true elevation masks. The U-Net interpolations are compared to the invalid images with yellow regions of NaNs.

Notice the similarity between the true and predicted elevation masks, the second and third columns, respectively. In the third case of invalid data shown, notice that the yellow patch of NaNs is near the top left of the elevation mask. For images like this, interpolation was only needed for a small number of pixels. On the other hand, for images containing mostly NaNs like the fourth case of invalid data shown, most of the elevation mask was reconstructed using the U-Net predictions.
2.5 Geocentric Pose Model

Figure 8: The U-Net Elevation Model contained 7,076,370 parameters.

After training an accurate elevation model, we assembled the Geocentric Pose Model, the second model in the ensemble framework. We concatenated the RGB images with their predicted elevation masks and fed these as inputs to predict the scale and angle of each image. Together, the U-Net Elevation Model and the Geocentric Pose Model predicted all three components of geocentric pose.

The Geocentric Pose Model was a convolutional neural network with a series of five double convolution layers and maximum pooling in between each series. The resulting shape was flattened and fed through six fully connected layers to generate the predictions. Batch Normalization and ReLU activation were used throughout the model. The model was configured with an 80-20% train-validation split and was trained for 800 epochs using Adam optimization, a learning rate of 0.001, and a mini batch size of 32.

2.5.1 Geocentric Pose Model Performance

Figure 9: text here

The learning curves of the Geocentric Pose Model were plotted on a logarithmic scale because the model learned so rapidly during the first 50 epochs. The plot on the right reveals a plateau in the validation MAE, but a continual
decrease in train MAE. However, the model did not overfit, as the validation loss did not start to increase. The model achieved an $R^2$ metric of 0.997 on the train set and 0.904 on the validation set.

![Figure 10: Metrics and plots to gauge the performance of the Geocentric Pose Model.](image)

**True vs. Predicted Value**  The predicted scale and angle are plotted against the true scale and angle. The linear regression fit reveals a slope of 1.006 and 1.008, respectively, which shows the accuracy of the model. Though it may seem that many points lie away from the regression line, there are only about 50 points that deviate, out of 1,185 points in total, hence showing the precision of the model.

**Performance Distributions**  The distributions of the true values and predicted values have similar shape and spread for both scale and angle, showing the model’s ability to learn trends between the RGB and elevation inputs and geocentric pose.

**Distribution of Absolute Error Loss**  The distributions of absolute error loss for both scale and angle are heavily skewed right, which shows that the model very rarely makes predictions that are faulty to a large degree. The distribution for the angle is more skewed than that of the scale.

**Sorted Absolute Error**  The graphs of sorted absolute error for both scale and angle take a skewed shape. On average, the model was able to predict the scale to within 0.00427 pixels per centimeter, and the angle to within 0.102 radians (5.84°).

### 2.6 Geocentric Pose Ensemble Model Metrics Summary

| Metric | U-Net Elevation Model | Geocentric Pose Model |
|--------|-----------------------|-----------------------|
|        | **Elevation (m)**     | **Scale (px/m)**      | **Angle (rad)**    |
| MAD    | Train 0.263 Validation 0.335 | Train 0.100 Validation 0.427 | Train 0.028 Validation 0.102 |
| MAE    | 0.847 1.049           | 0.373 0.755           | 0.135 0.214        |
| $R^2$  | 0.926 0.865           | 0.995 0.864           | 0.999 0.943        |
3 Discussion

3.1 Insights from Deep Learning Models

Autoencoder Model  The Optimal Performance of the Autoencoder validated our Deep Learning approach to Geo-centric Pose. The latent space representation that the model used was twelve times smaller than the original image yet preserved important features within the data, shown through the similarity between the original images and the autoencoded images. The visualization of the latent space representation through Multi-Dimensional Scaling revealed correlations in angle, with higher-angle images grouped on the right and lower-angle images grouped on the left. No clear correlation was determined when observing the grouping of images in scale, though this may be due to the low variance retained when reducing dimensionality by such a large factor. Still, the correlations seen in the visualizations motivate a future avenue of exploration, which involves using the autoencoder as a feature extractor and feeding outputs into either the Geocentric Pose or Elevation model.

U-Net Elevation Model  We experimented with the usefulness of the features extracted by the elevation model by training two different Geocentric Pose models. The first model, which was our control, only received the RGB images as inputs. The second model received RGB images and elevation predictions from the U-net model as inputs. Both models were trained using the same architecture and the same hyperparameters, with the exception of the number of input channels.

The performance of the model trained on the elevation predictions exceeded that of the control model with a statistically significant p-value of $2.07 \cdot 10^{-4}$. We believe that such great improvement was possible because the elevation of buildings is useful to predict the scale, which relies on a measure of the conversion factor between pixels in the image and distances in the real-world. The elevation may have also been beneficial in predicting angle, as knowledge of the building heights can help in determining the orientation of satellites.

Geocentric Pose Model  The architecture of the Geocentric model was crucial to its performance. Several smaller models with less than one million parameters were experimented with, and different types of models such as fully convolutional neural networks were also attempted. However, these models underfitted the data and could not achieve an MAE less than 0.8 for both scale and angle. Thus, we concluded that the task at hand was sufficiently complex as to require a very deep network with greater than five million parameters; therefore, we focused our efforts towards this area, ultimately choosing a similar architecture that Sun, C. (2022) used for wildfire detection\[5\].

When analyzing specific predictions of low accuracy, we were surprised to find that the model struggled the most for predicting the angle of some near-nadir images, likely because the height of buildings are difficult to perceive at these angles, making it difficult to distinguish between angles in this range.

Limitations and Future Improvements  As stated previously, computational constraints obligated us to downsize the input images, which lowered the resolution, making it more difficult to predict geocentric pose. Instead, improvements can be made by employing random cropping, which would allow us to reduce the number of parameters while keeping the original resolution. Performance can further be enhanced through the use of image augmentation, particularly for cities such as Atlanta and Jacksonville, to increase the resilience of the model towards distortions of the image and to prevent the model from overfitting on the city with the most labeled data. Additionally, both the Geocentric Pose and the U-net Elevation Model may benefit from the usage of Ground Sample features included in the meta-data set and/or the city name.

3.2 Comparison to Prior Research

While Christie et al. achieved a cumulative $R^2$ metric of 0.799, the models designed in this study achieved a cumulative $R^2$ metric of 0.891, demonstrating a major increase in accuracy and reliability of geocentric pose predictions. We attribute this higher performance to the ensemble methodology and the data preprocessing that served as quality control.
4 Conclusion

With a reliable ensemble model to identify the geocentric pose of both near-nadir and oblique images, the versatility of software fueling natural disaster response can be enhanced. We have designed an accurate framework that has applications in disaster response through vision-aided navigation, change detection, map alignment, and search-and-rescue. When UAVs learn the physical features of their surroundings, they can transfer information between each other to allow for specialized systems and more effectively summon human users when assistance is needed.

Need for Further Research Further research is necessary to investigate how the model architectures described in this paper could be improved. Changing the size of the bottleneck layer in the U-Net Elevation Model or using two separate, simpler models to predict scale and angle are avenues that should be explored. Understanding how the resolution of images impacts both the Elevation and the Geocentric Pose Model could lend insights into how to improve feature detection using the Autoencoder. In addition, further testing needs to be performed to understand how well both the Geocentric Pose and the U-Net Elevation Model generalize to new data from cities around the world, since different city architectures could pose a potential challenge for the models.

References

[1] Mishra, B., Garg, D., Narang, P., & Mishra, V. (2020). Drone-surveillance for search and rescue in natural disaster. Computer Communications, 156, 1-10.

[2] Foster, K., Christie, G., & Brown, M. (2020). Urban Semantic 3D Dataset. IEEE Dataport. https://dx.doi.org/10.21227/9frn-7208

[3] Christie, G., Foster, K., Hagstrom, S., Hager, G. D., & Brown, M. Z. (2021). Single view geocentric pose in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1162-1171).

[4] Christie, G., Abujder, R. R. R. M., Foster, K., Hagstrom, S., Hager, G. D., & Brown, M. Z. (2020). Learning geocentric object pose in oblique monocular images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 14512-14520).

[5] Sun, C. (2022). Analyzing Multispectral Satellite Imagery of South American Wildfires Using CNNs and Unsupervised Learning. arXiv preprint arXiv:2201.09671