

Looking for Homes from Space

A Hybrid ViT - U-Net Model for Building Segmentation from Satellite Imagery

Final Project Report

Irti Haq, Youngkyu Ko, Xunan Li

Table of Contents

MOTIVATION	2
RESEARCH QUESTION	2
RELATED WORK.....	2
OUR DATA.....	3
EVALUATION METRICS.....	3
OUR APPROACH.....	4
TRANSUNET	4
VIT-UNET HYBRID: OUR IMPLEMENTATION OF TRANSUNET	8
KEY CHANGES MADE.....	8
DATA AUGMENTATIONS	10
OUR FINAL MODEL PARAMETERS.....	11
RESULT	12
MODEL PREDICTIONS	15
EXPERIMENT: COMPARING DIFFERENT LEVELS OF AUGMENTATION.....	19
DISCUSSION OF FINDINGS AND INSIGHTS	22
LIMITATIONS AND FUTURE WORK.....	22
ETHICAL CONSIDERATIONS	23
CONCLUSION	23
WORKS CITED.....	24

Motivation

Access to accurate, up-to-date maps of building is essential for disaster response, population estimation, infrastructure development, and humanitarian aid. Yet many regions, especially rural or underserved areas, lack such data. Recent advances in high-resolution satellite imagery and deep learning provide a promising path forward. While prior work has mainly used CNN-based models (e.g., U-Net), Vision Transformers (ViTs) have shown promising results by using Attention they are able to better capture global features dependencies. This motivates our investigation into a hybrid ViT + U-Net architecture to robustly detect buildings across diverse regions.

Research Question

“Can a Hybrid ViT + U-Net model achieve high segmentation accuracy and low boundary error for building segmentation across diverse urban environments?”

Related Work

Previous work in building segmentation from satellite imagery has largely relied on CNN-based architectures such as U-Net, which excels at preserving spatial detail. Vision Transformers (Dosovitskiy et al., 2020) have recently emerged as powerful alternatives due to their ability to model global context, although they are computationally intensive. TransUNet (Chen et al., 2021) demonstrated a successful hybrid of ViT and U-Net in the medical domain, combining the strengths of both architectures. In the context of remote sensing, Mou et al. (2022) reviewed the use of ViTs and emphasized the potential of hybrid approaches. Our work extends these insights into the geospatial domain, drawing on the DeepGlobe Challenge (Demir et al., 2018), which established benchmarks for satellite-based building detection.

U-Net: Strong local detail segmentation model, widely used in biomedical tasks.

ViT (Dosovitskiy et al., 2020): This foundational paper introduced Vision Transformers and their ability to outperform CNNs on image classification tasks, motivating their application to dense prediction problems like segmentation

TransUNet (Chen et al., 2021): Hybrid of ViT and U-Net, effective in medical segmentation.

DeepGlobe Challenge (Demir et al., 2018): Benchmarks CNN-based building detection from satellite images.

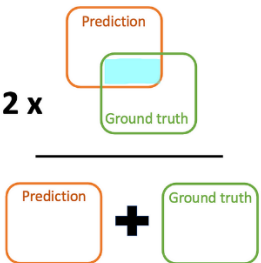
Bazi et al. (2021): Reviews Vision Transformer use in remote sensing, stressing the value of hybrid models.

Our Data

For our model training and evaluation, we use the GaoFen-7 (GF-7) dataset, which includes high-resolution satellite imagery and corresponding building segmentation masks from six representative cities in China. The dataset consists of 5175 total image-mask pairs, divided into 3106 training, 1034 validation, and 1035 testing samples. Each image is resized to 224×224 pixels and split into 16×16 patches for Transformer processing. The dataset features binary classification (building vs. background) and serves as a solid foundation for evaluating segmentation performance. To improve generalizability of our model, we applied moderate data augmentations using the Albumentations library. These included geometric transformations such as rotations and flips, as well as color adjustments like brightness and contrast modifications. We deliberately avoided overly aggressive augmentations, which had previously led to performance degradation.

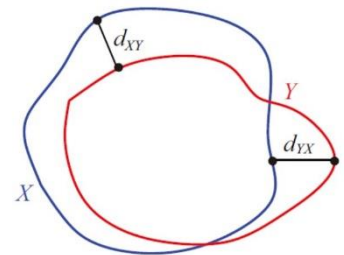
Evaluation Metrics

To evaluate segmentation Quality, we employed two metrics: Dice Score (F1) and HD95 (Hausdorff Distance at the 95th percentile). The Dice Score quantifies the overlap between the predicted and ground truth masks, ranging from 0 (no overlap) to 1 (perfect match). It is particularly useful for imbalanced datasets where the building area may be small relative to the background.

$$\text{Dice} = \frac{2 \times \text{Area of overlap}}{\text{Total area}} = \frac{2 \times \text{Area of overlap}}{\text{Prediction} + \text{Ground truth}}$$


HD95, on the other hand, measures the boundary error between predicted and actual building contours. It provides robustness against outliers by focusing on the 95th percentile rather than the maximum deviation. These complementary metrics help us assess both region-wise accuracy and boundary precision.

$$d_H(X, Y) = \max\{d_{XY}, d_{YX}\} = \max\left\{\max_{x \in X} \min_{y \in Y} d(x, y), \max_{y \in Y} \min_{x \in X} d(x, y)\right\}$$



Our Approach

TransUNET

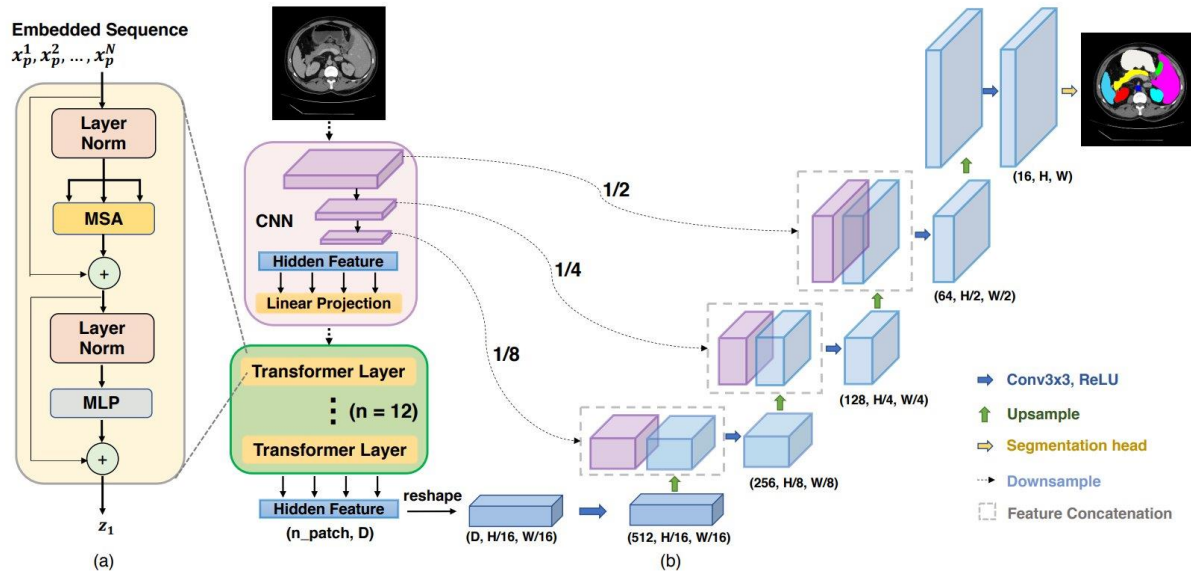


FIGURE 1: TRANSUNET ARCHITECTURE BY CHEN ET AL., 2021

TransUNet is a state-of-the-art medical image segmentation developed by Chen et al (2021) that merges the benefits of a Transformer and the U-Net architecture. The Model combines a CNN network (ResNet) and transformer blocks as the encoder and upsampling layers as the decoder. It is designed to capture both global, long-range dependencies (via Transformers) and local, fine-grained details (via U-Net). Our project builds on Chen et al (2021) work and adopts the TranUNet architecture for satellite image segmentation.

Historically, fully convolutional networks (FCNs) such as U-Net have dominated medical image segmentation tasks and achieved remarkable success due to their strong representational power. However, their performance is limited by the intrinsic locality of convolution operations, which hinders their ability to explicitly model long-range dependencies. As a result, in medicine for example these architectures often struggle with segmenting target structures that exhibit substantial inter-patient variability in texture, shape, and size. (Chen et al., 2021)

In contrast, Transformers—originally designed for sequence-to-sequence prediction—leverage global self-attention mechanisms and excel at modeling global context and capturing long-range dependencies. They have demonstrated superior transferability in downstream tasks, particularly when trained on large-scale datasets. However, they have limited localization ability due to insufficient low-level details.

Chen et al (2021) found that combining a transformer (for encoding tokenized image patches) with naïve upsampling of the hidden feature representations into a dense full resolution output couldn't produce satisfactory result due to the fact that the transformer treated the input as 1D sequences and exclusively focus on modeling the global context at all stages. This led to low resolution features that lacked enough detailed localization information (i.e. precisely identify and delineate the exact boundaries and fine-grained spatial details of objects or structures) for accurate segmentation. This means they can identify what an object is and where it generally is, but not its precise shape or boundary. This information can't be effectively recovered using direct up sampling to the full resolution and led to the loss of low-level details and feature

resolution and coarse segmentation masks. CNN's conversely excel extracting the low-level visual cues necessary to capture these intricate spatial details (Chen et al., 2021).

TransUNet aims to solve both the long-range dependencies and global context problem of CNN's by establishing a self-attention mechanism from the perspective of sequence-to-sequence prediction and by employing a hybrid CNN-Transformer architecture it is able to compensate for the loss of feature resolution brought by Transformers by leveraging both detailed high-resolution spatial information from CNN features and the global context encoded by Transformers. Building on the U-NET Architecture TransUNet upsamples the self-attentive feature encoded by Transformers and combines them with different high-resolution CNN features skipped from the encoding path in order to enable precise localization. Chen et al (2021) found that more intensive incorporation of low-level features resulted in overall better segmentation accuracy. (Chen et al., 2021)

To better understand how TransUNet achieves this balance between local detail and global context, it is useful to break down its core architectural components further. Starting with the encoder, Transunet employs two key components a CNN network (ResNet) and transformer blocks. Chen et al (2021) found that this hybrid approach performs better than simply using a pure Transformer as the encoder.

ResNet: ResNet-50 serves as the encoder backbone used as a feature extractor to generate a feature map which are used to form patches as the input of the vision transformer to encode. ResNet performs downsamplings and extracts high-level features. Using skip connections, ResNet is able to alleviate the vanishing gradient problem. TransUNet also make one additional to ResNet though the use of [bottleneck blocks](#) within ResNet. These blocks apply a 1x1 convolution to reduce the number of channel, then a 3x3 convolution, and another 1x1 convolution to get back to the original dimensions. This design achieves same thing result as standard ResNet blocks but with fewer parameters, and thus reducing training time. Furthermore, [group normalization](#) is used instead of batch normalization to stabilize training when batch sizes are small. Unlike batch normalization, group normalization is independent of batch size, making it more robust and accurate under such conditions.

TransUNet uses the following structure for ResNet-50:

- **Input:** The ResNet-50 takes input images that are **224x224 pixels with 3 color channels**.
- **First Convolution Layer:** Uses 64 7x7 kernels with stride 2 with output of (64, 112 X 112).
- **Max Pooling:** resulting images are in the size of 64, 56x56.
- **Bottleneck Networks:** The core of the ResNet-50 consists of three successive "bottleneck" networks, each progressively reducing the spatial dimensions of the image while increasing the number of features.
- Each of the three blocks also outputs the hidden feature for a skip connection to the decoder side.
- **"The rest of the ResNet-50 model consists of three successive bottleneck networks.**
 - The first part contains three bottleneck blocks, and the resulting images are in the size of (256, 56, 56).
 - The second part consists of four bottleneck blocks, and the resulting images are in the size of (512, 28, 28).
 - The last part consists of nine bottleneck blocks, and the resulting images are in the size of (1024, 14, 14)" (Zhang & Tan, n.d.)
 - **Note This section was copied from <https://tianjinteda.github.io/Transunet.html>**

Layer (type:depth-idx)	Param #
Transformer: 1-1	--
└─Embeddings: 2-1	--
└─ResNetV2: 3-1	--
└─Sequential: 4-1	--
└─StdConv2d: 5-1	9,408
└─GroupNorm: 5-2	128
└─ReLU: 5-3	--
└─Sequential: 4-2	--
└─Sequential: 5-4	--
└─PreActBottleneck: 6-1	75,008
└─PreActBottleneck: 6-2	70,400
└─PreActBottleneck: 6-3	70,400
└─Sequential: 5-5	--
└─PreActBottleneck: 6-4	379,392
└─PreActBottleneck: 6-5	280,064
└─PreActBottleneck: 6-6	280,064
└─PreActBottleneck: 6-7	280,064
└─Sequential: 5-6	--
└─PreActBottleneck: 6-8	1,512,448
└─PreActBottleneck: 6-9	1,117,184
└─PreActBottleneck: 6-10	1,117,184
└─PreActBottleneck: 6-11	1,117,184
└─PreActBottleneck: 6-12	1,117,184
└─PreActBottleneck: 6-13	1,117,184
└─PreActBottleneck: 6-14	1,117,184
└─PreActBottleneck: 6-15	1,117,184
└─PreActBottleneck: 6-16	1,117,184
└─Conv2d: 3-2	787,200
└─Dropout: 3-3	--

THE RESNETV2 COMPONENT FROM OUR TRANSUNET IMPLEMENTATION."

- **Output:** Resnet 50 takes images of 224x224 in 3 channels and outputs 14x14 feature maps in 1024 channels. The images are then feed into the Vision Transformer where each pixel is an image patch with a total of 196 patches (Chen et al., 2021; Zhang & Tan, n.d.)

Embedding Layer: Between the ResNet50 and Transformer layers there is a 2D convolution layer then reduces the number of channel from 1024 to 768. These refined image patches are then flattened and transposed into a 2D matrix of size (196 x 768), serving as the image embeddings. Finally, before these embeddings are fed into the Transformer model for encoding, position embeddings are added, and dropout is applied

Transformer: The second key part of the TransUNet encoder is the 12 stacked Transformer Block. These are used to tokenize pathways and extract abstract features from the original input and consequently the global context. (Ji et al., 2024).

The transformer encoder block consists of two components. **Multihead Self-Attention (MSA) layer** with layer normalization consisting of a matrix of size (196, 196) where each element represents the similarity between two image patches and with a SoftMax function to get the weights of all image patches given each image patch. The concatenated the results of all 12 heads of the MSA are then fed into a **Feedforward Multi-Layer Perceptron (MLP) layers** with layer normalization. (Chen et al., 2021). The Self-Attention layers help the model know where to

focus. Both the MSA and MLP layers also incorporate skip connections (residual connections). The output of the 12 stacked Transformer Block is the encoded embeddings for an input image denoted as a hidden feature matrix with dimensions (196, 768) in figure 1. (Zhang & Tan, n.d.) This means that each of the 196 image patches is represented by a 768-dimensional feature vector that captures both local details and long-range dependencies. In other words, it captures and extracts the essential global contexts and high-level abstract semantic features from the image, effectively transforming complex visual data into a form that highlights the most relevant patterns. These embeddings serve as a rich, globally contextualized representation of the input image, ready to be decoded into a segmentation map.

Decoder: Moving on the Decoder part of the model, the decoder reshapes the output hidden feature matrix from the encoder and performs multiple Cascade up sampling (CUP) blocks until the full resolution achieved. Each step typically involves a 2x upsampling operation, followed by a 3×3 convolution layer and a ReLU activation successively (Chen et al., 2021). During this the decoder also integrates high-resolution CNN feature maps from the ResNet encoder via skip connections at different resolution levels. (Ji et al., 2024)

The use of Skip Connections in the U-Net Structure is a critical aspect of the decoder and is fundamental to TransUNets performance as it allows the model to benefit from the global Context captured by the Transformer while also benefiting from the precise low-level spatial details extracted by the ResNet CNN at various different resolution level, which is essential for accurate boundary detection capabilities. It essentially feeds back in fine-grained spatial details and low-level features that were lost by the transformer and allows the decoder to use both the low-level features from the CNN layers and high-level features from the Transformer and fusing them together

Finally the model applies the segmentation head, a final convolution that change the channel number to be equal to the number of classes (9 in the case of TransUNET but only 2 in our building segmentation case) Each pixel has one channel for each class. These channels represent the probability of that a pixel belongs to a certain class after applying softmax. (Zhang & Tan, n.d.)

Pretrained Model Weights: TransUNet leverages pre-trained Weights from ResNet-50 ViT-B_16 Model developed by Researchers at Google to enhance its performance and to reduce training time. ResNet-50 ViT-B_16 is used as part of the hybrid encoder. Both the ResNet-50 Component and the Vision Transformer were pre-trained on a large-scale ImageNet dataset. This use of pre-trained Weights allows the model to benefit from robust, learned visual representations, which are then fine-tuned for specific medical image segmentation tasks for TransUNet and in our case Satellite Image Building Segmentation. **The decoder on**

```
def forward(self, x):
    # Self-Attention Layer
    h = x
    x = self.attention_norm(x)
    x, weights = self.attn(x)
    x = x + h

    # MLP Layer
    h = x
    x = self.ffn_norm(x)
    x = self.ffn(x)
    x = x + h
    return x, weights
```

FIGURE 3 CODE SNIPPET SHOWING TRANSFORMER BLOCK FROM OUR TRANSUNET IMPLEMENTATION FROM BLOCK CLASS

the other hand is trained from scratch. This allows the Decoder to adapt to the specific tasks while still allowing the encoder to benefit from general-purpose visual representations learned on large scale image datasets.

ViT-UNet Hybrid: Our Implementation of TransUNet

Our Model builds on Chen et al. (2021) TransUNet by adapting it for the use of Building Segmentation from Satellite Imagery from Chinese GaoFen-7 (GF-7) satellite. Most of the Code and Architecture for the Model has been directly taken from the TransUNet project by Chen et al. (2021). The primary goal is to achieve high segmentation accuracy (Dice score) and low boundary error (HD95) across diverse urban areas.

Key Changes Made

Overall, in our effort to adapt and optimize the TransUNet model for Building Segmentation from Satellite Imagery we made several key to the original TransUNet Model.

Optimizer

Originally TransUNet initially came with Stochastic Gradient Descent with momentum as its original optimizers. Throughout our training process we wanted to experiment with more advanced optimizers. We first moved to Adam then moved to AdamW which is a variant of the Adam optimizer that decouples weight decay from gradient updates, leading to better generalization in deep learning models. Going from SGD to Adam and then AdamW greatly improved convergence and helped to significantly improve training times. Seeing this we then combined AdamW combined with AMSGrad. AMSGrad is a modification of Adam that ensures stable convergence by using the maximum of past squared gradients instead of an exponentially weighted average (Gugger & Howard, 2018). This had an even greater effect in helping improved convergence, reduce training time and allowed us to get better results in fewer epochs

Learning Rate Scheduler

We also made key changes to the Learning Rate Scheduler. The Original TransUNet model using a **polynomial learning rate decay** However this led to inconsistent unstable convergence behavior and slower training progress, particularly in the early epochs. In order to improve this we replaced it with **CosineAnnealingWarmRestarts**, A learning rate scheduler that periodically reduces the learning rate following a cosine curve and then restarts it, helping models escape local minima and improve convergence. This led to a Smoother, more adaptive learning rates & resulted in faster convergence, reduced training time, and improved overall stability

Data Augmentation

When it came to data augmentation, we experiment with different levels augmentations. While our initial approach using strong augmentations yielded extremely poor results. Through iterative refinement, for our final model went with a more moderate data augmentation pipeline using Albumentations. This pipeline

```
=====
Layer (type:depth-idx)
=====
|Transformer: 1-1
|  |Embeddings: 2-1
|  |  |ResNetV2: 3-1
|  |  |Conv2d: 3-2
|  |  |Dropout: 3-3
|  |Encoder: 2-2
|  |  |ModuleList: 3-4
|  |  |LayerNorm: 3-5
|DecoderCup: 1-2
|  |Conv2dReLU: 2-3
|  |  |Conv2d: 3-6
|  |  |BatchNorm2d: 3-7
|  |  |ReLU: 3-8
|  |ModuleList: 2-4
|  |  |DecoderBlock: 3-9
|  |  |DecoderBlock: 3-10
|  |  |DecoderBlock: 3-11
|  |  |DecoderBlock: 3-12
|SegmentationHead: 1-3
|  |Conv2d: 2-5
|  |Identity: 2-6
=====
Total params: 105,125,538
Trainable params: 105,125,538
Non-trainable params: 0
=====
```

FIGURE 4 HYBRID ViT UNET MODEL

applies a combination of geometric transforms, including rotation and flips, alongside various color augmentations such as brightness, contrast, and color shifting, and the addition of multiplicative noise. This balanced approach proved more effective for our specific task and led to a more generalizable model. We will discuss the results of our experimentation with Lighter Geometric Augmentations compared to our more moderate data augmentation pipeline on model generalizability later in the report.

Tunning

Finally, extensive tuning was performed across several key hyperparameters. This involved optimizing the batch size, learning rate, training epochs, and the configuration of skip connections, all of which were critical for achieving the final model performance.

Data Augmentations

MODERATE DATA AUGMENTATION PIPELINE (USED FOR OUR FINAL MODEL)

RandomRotate90: Rotates by 90, 180, or 270 degrees with a **0.5 probability**.

HorizontalFlip: Flips horizontally with a **0.5 probability**.

VerticalFlip: Flips vertically with a **0.5 probability**.

SomeOf (applies 2 of the following with a **0.8 probability**):

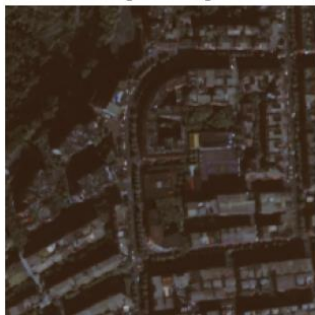
- **RandomBrightnessContrast:** Adjusts brightness and contrast within a **0.1 limit**.
- **RGBShift:** Shifts R, G, and B channels within an **8 limit**.
- **HueSaturationValue:** Shifts hue within a **5 limit**, saturation within an **8 limit**, and value within a **5 limit**.
- **RandomGamma:** Applies gamma correction within a **(90, 110) range**.

OneOf (applies 1 of the following with a **0.4 probability**):

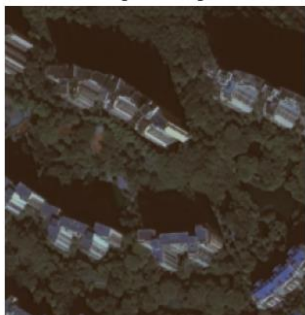
- **MultiplicativeNoise:** Applies multiplicative noise with a **multiplier range of (0.7, 1.2)**, per-channel, and elementwise.

Examples

Original Image



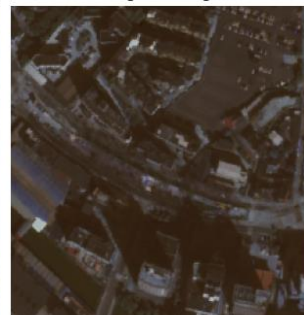
Original Image



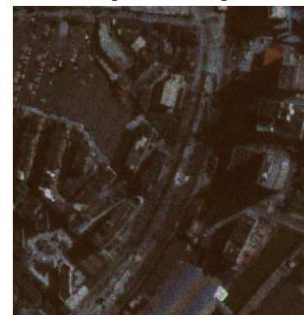
Augmented Image



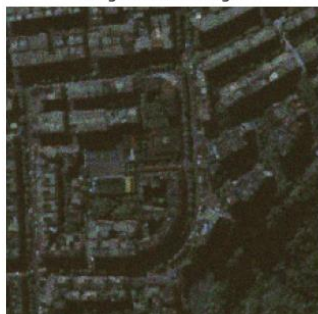
Original Image



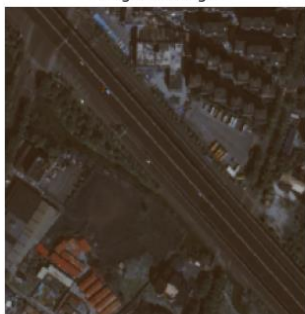
Augmented Image



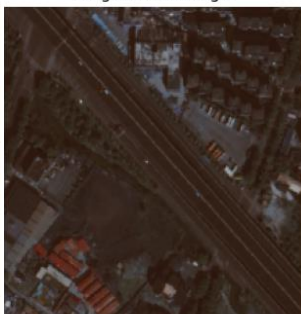
Augmented Image



Original Image



Augmented Image



Original Image



Augmented Image



GEOMETRIC TRANSFORMS ONLY DATA AUGMENTATION PIPELINE

RandomRotate90: Rotates by 90, 180, or 270 degrees with a **0.5 probability**.

HorizontalFlip: Flips horizontally with a **0.5 probability**.

VerticalFlip: Flips vertically with a **0.5 probability**.

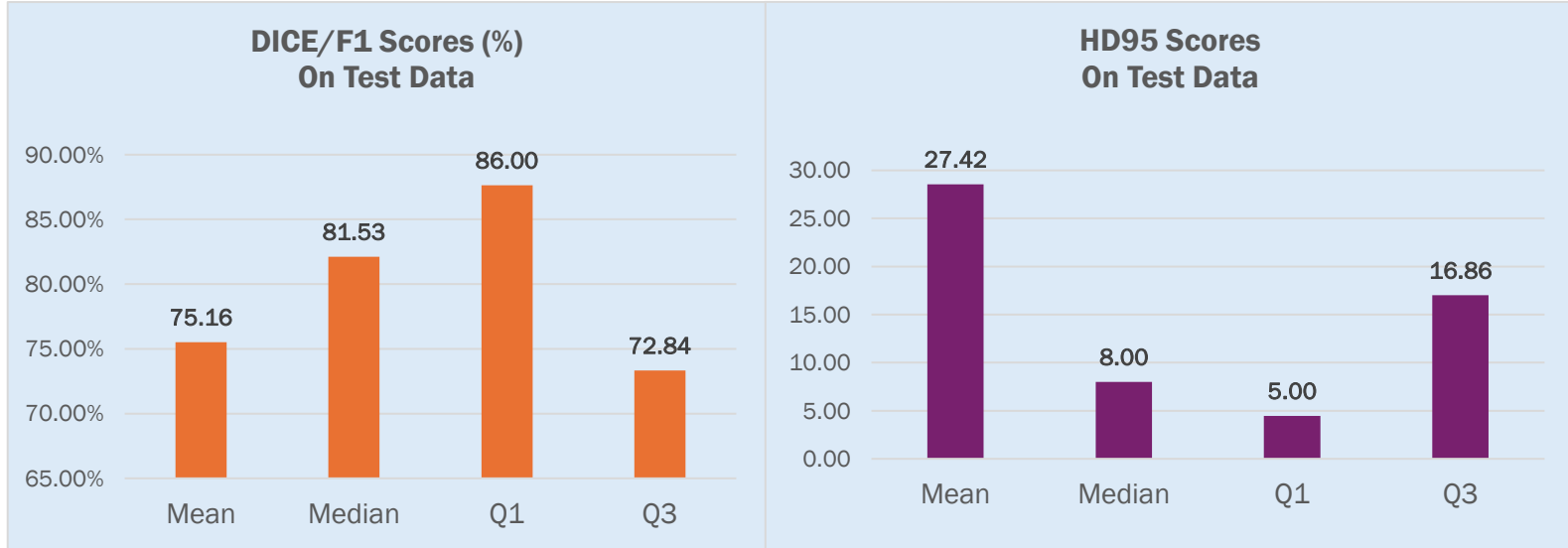
Our Final Model Parameters

- **Dataset:** Chinese GaoFen-7 (GF-7) satellite imagery dataset.
 - **Total Images: 5175** = **Training:** 3106 images + **Validation:** 1034 images + **Test:** 1035 images
- **Number of Classes** = 2 Building vs. Background
- **Image Size** : 224 X 224 , **Patch Size** = 16 X 16
- **Augments Used:** [Moderate Augmentation Pipeline](#) with Geometric Transforms, Color Augments, & Multiplicative Noise
- **Device:** GPU (Cuda)
- **Optimizer:** AdamW + AMSGrad
- **Learning Rate Scheduler:** CosineAnnealingWarmRestarts
- **Base Learning Rate:** 0.001
- **Loss Function:** Dice + CrossEntropy (CE)
- **Batch Size:** 25
- **Number of Epochs:** 94

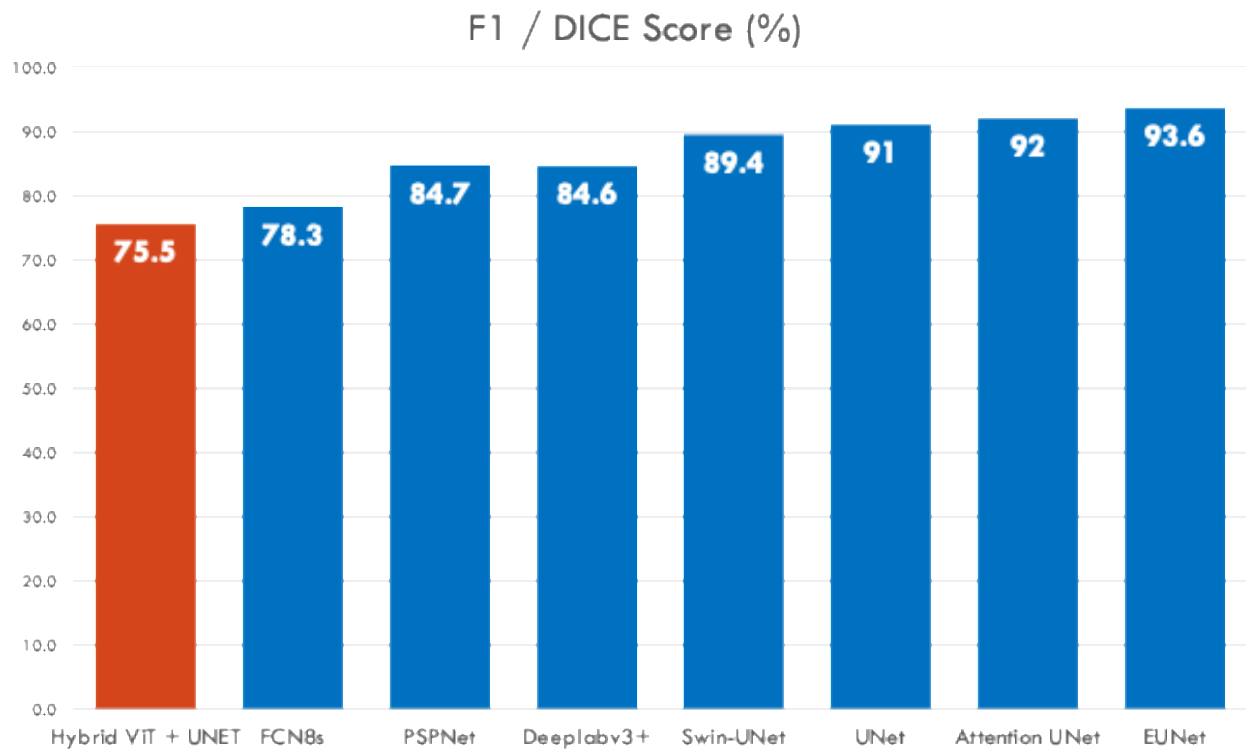
Result

Our model's performance on the GaoFen 7 (GF7) test sets, including Dice Score and HD95, is detailed below.

HYBRID VIT-UNET MODEL'S FINAL TEST SCORES ON GF7 TEST DATASET



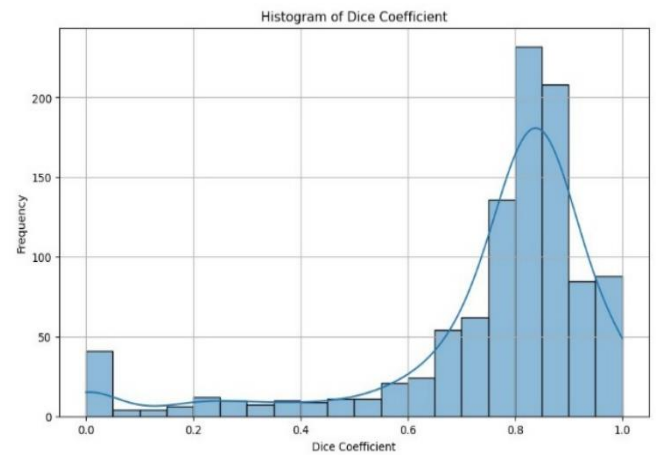
COMPARING OUR DICE / F1 SCORE AGAINST OTHER MODELS



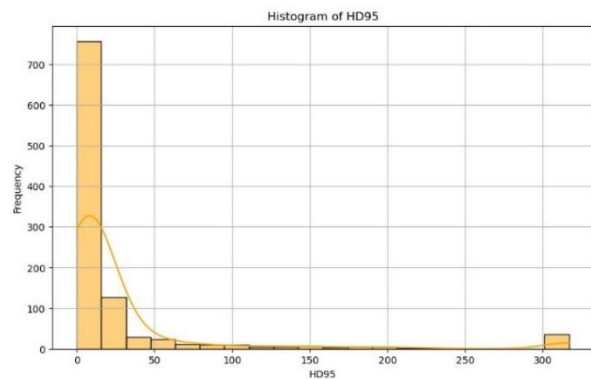
Source: Han, R., Fan, X., & Liu, J. (2024). EUNet: Edge-UNet for Accurate Building Extraction and Edge Emphasis in Gaofen-7 Images. Remote Sensing, 16(13), 2397.

Compared to Other Models Tested on the GF7 Dataset our model performs relatively well, it's up there in a similar range. It's only 3 percentage point lower than FCN8 Model. Overall, the results are promising especially as a proof of concept

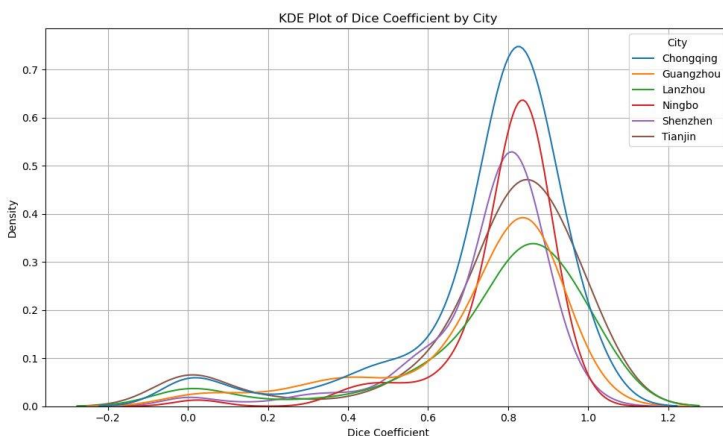
Our model demonstrated strong performance in terms of segmentation accuracy, especially in dense urban areas. On the test set, the Dice score distribution peaked around 0.8, with a Left-skewed shape and a long tail of error indicating that most predictions had high overlap with ground truth masks. A small number of predictions achieved a perfect Dice score of 1, while a few outliers scored near zero, suggesting poor predictions for edge cases, such as patches with no buildings or occlusions, this is also indicated by Small Bump in the Dice Coefficient at Zero suggesting there might be a small subset of images where the model has extremely poor performance



The HD95 metric showed a Right-skewed distribution, with a long tail of error. The majority of predictions achieving boundary errors within 10 pixels. However, some extreme cases exhibited HD95 values above 100 pixels, highlighting the model's limitations in edge delineation under complex conditions. These errors were often observed in rural or ambiguous regions where visual cues were sparse. There is also a Bump at 316.8, the max distance on of an image, this was a change we made to calculation of HD95 Scores so that we it detects a False Positive in images where aren't buildings, HD95 is set to the max distance. This was done to penalize false positives more strongly. In the original TransUNet score they set it Zero for false postives.



Comparing the results of the models with and without data augmentation confirmed the benefits of moderate augmentations on generalizability. Specifically, models trained with geometric and color-based augmentations outperformed those trained with only geometric transformations. We delve into the details later in the report. Additionally, comparison with other published models on the same dataset, such as EUNet, indicated that our hybrid approach was competitive and in line with the state-of-the-art models.



Finally, performance across the various different cities within the dataset was mostly consistent. For instance, Ningbo achieved the highest Dice scores at (78.5%) and Guangzhou (72.3%) had the lowest however difference between the two is pretty small, only about 5% indicating strong model performance in dense urban environments. Conversely, regions with more complex or sparse patterns, such as remote patches in the dataset, had significantly lower Dice scores and higher HD95 values, reflecting difficulties in accurately segmenting buildings with limited or noisy visual context. Despite these outliers, the overall consistency across urban centers suggests that the model generalizes reasonably well to different architectural styles and geographic layouts. This offers

promising potential for extending the model to other high-resolution satellite datasets, assuming similar image quality and annotation standards.

City	Dice Coefficient Mean	Dice Coefficient Std	HD95 Mean	HD95 Std
Chongqing	74.4%	22.3%	26.69	59.48
Guangzhou	72.3%	22.8%	29.42	56.69
Lanzhou	76.7%	25.0%	33.31	74.00
Ningbo	78.5%	14.7%	15.57	23.42
Shenzhen	74.3%	17.7%	23.65	51.72
Tianjin	75.3%	26.3%	34.51	79.73
Overall	75.2%	22.1%	27.42	61.27



MAP OF CHINA FOR REFERENCE WITH ALL OF THE CITIES MARKED

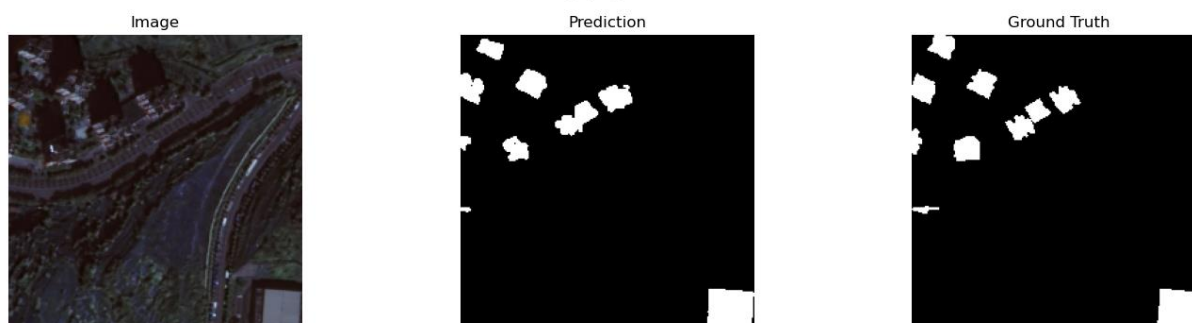
Model Predictions

Here are some examples of some of our predicated masks selected from the first and last 5 cases

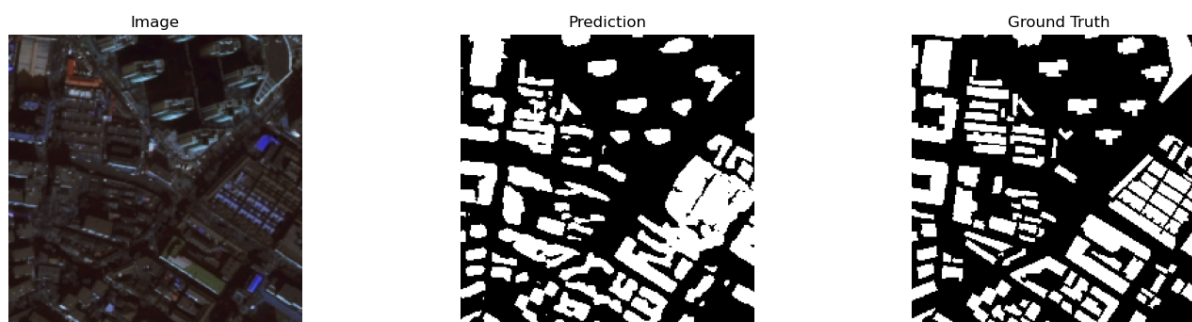
Prediction for Case 0 : Chongqing - Dice: 0.9358, HD95: 9.2195



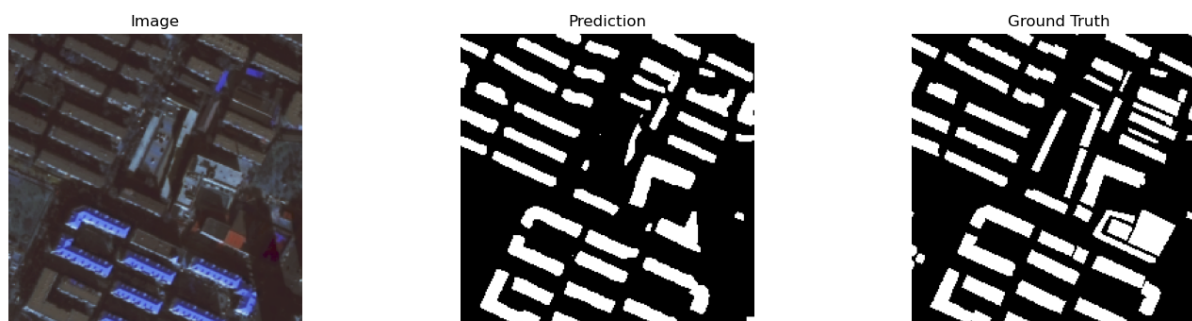
Prediction for Case 1 : Chongqing - Dice: 0.8951, HD95: 4.0000



Prediction for Case 1031 : Tianjin - Dice: 0.8000, HD95: 5.0000

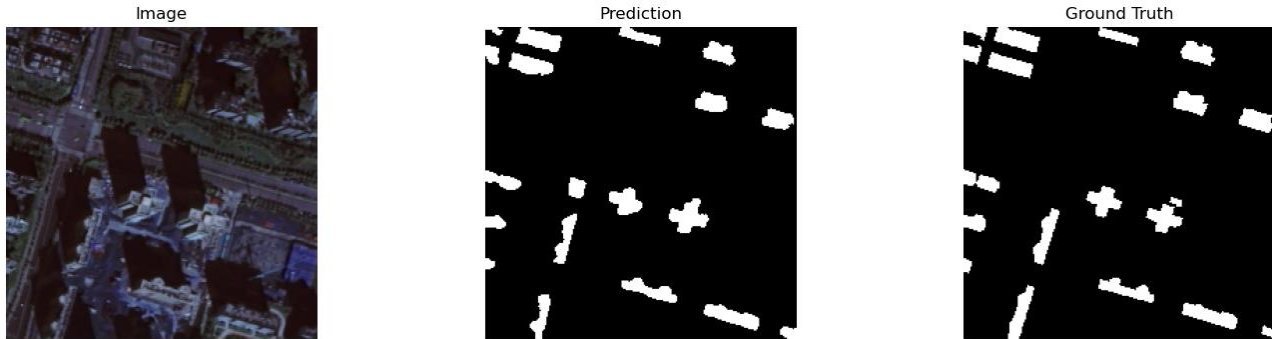


Prediction for Case 1032 : Tianjin - Dice: 0.8646, HD95: 5.0000



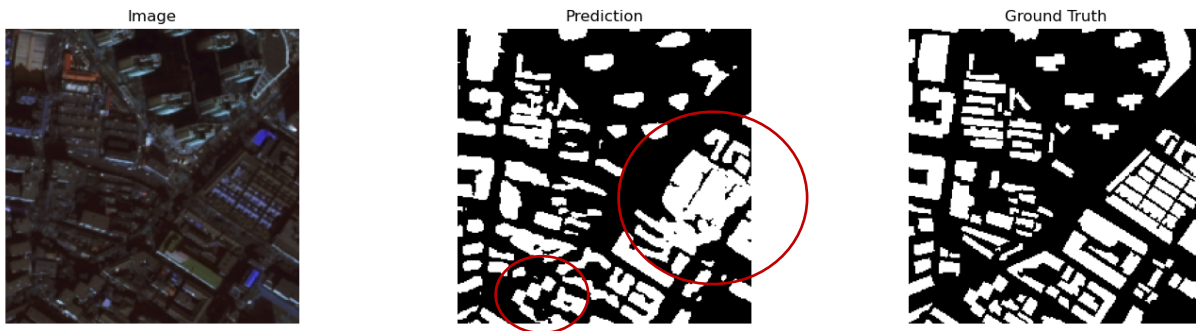
Overall Looking at the results we can see that for the most part the model is doing a pretty good job especially when we are dealing with simple building shapes and with regular repeating grid like patterns like for example in case 1032. Case 4 also does a good job illustrating this.

Prediction for Case 4 : Chongqing - Dice: 0.8767, HD95: 2.8284



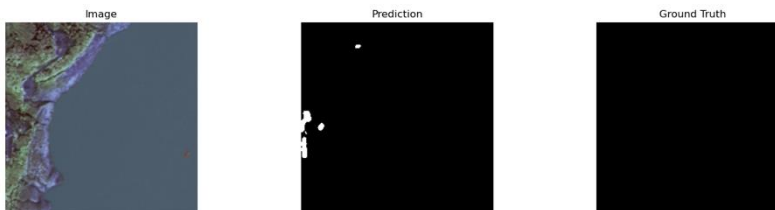
But sometimes it struggles capturing more fine-grained patterns like in case 1031. But as shown in the same case it does do a decent job with more irregular shapes like for example the three diamonds in the bottom of case 1031. We can also see that even when there isn't much contrast between building and background the model is performing decently.

Prediction for Case 1031 : Tianjin - Dice: 0.8000, HD95: 5.0000

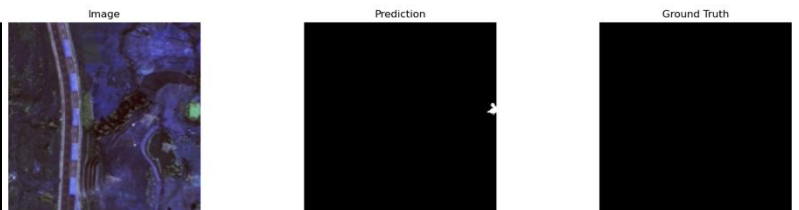


CASES WITH A DICE SCORE OF 0

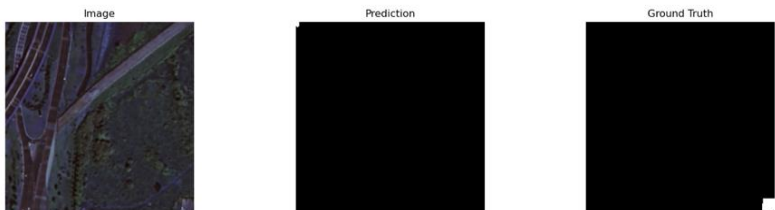
Worst Case 53 : Chongqing - Dice: 0.0000, HD95: 316.7838



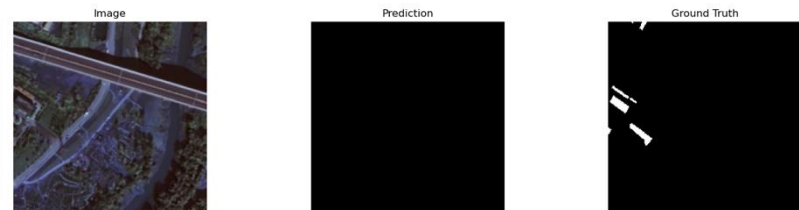
Worst Case 59 : Chongqing - Dice: 0.0000, HD95: 316.7838



Worst Case 66 : Chongqing - Dice: 0.0000, HD95: 306.8990

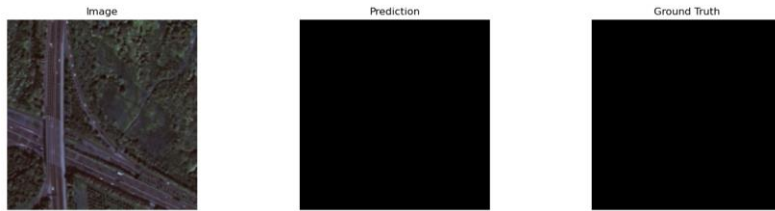


Worst Case 75 : Chongqing - Dice: 0.0000, HD95: 316.7838

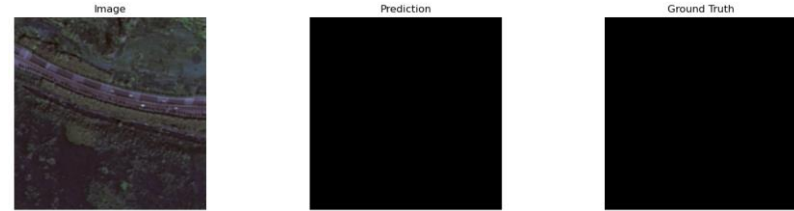


CASES WITH A DICE SCORE OF 1

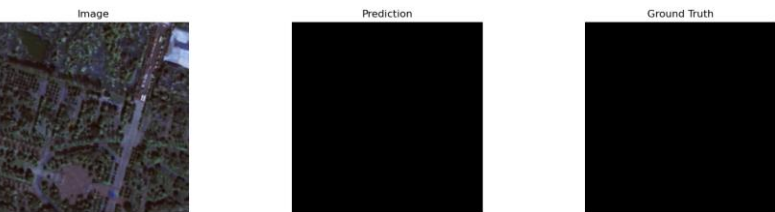
Best Case 6 : Chongqing - Dice: 1.0000, HD95: 0.0000



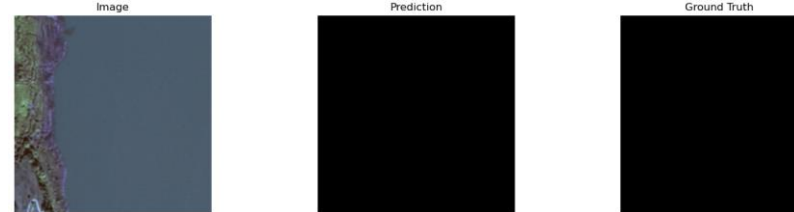
Best Case 23 : Chongqing - Dice: 1.0000, HD95: 0.0000



Best Case 13 : Chongqing - Dice: 1.0000, HD95: 0.0000

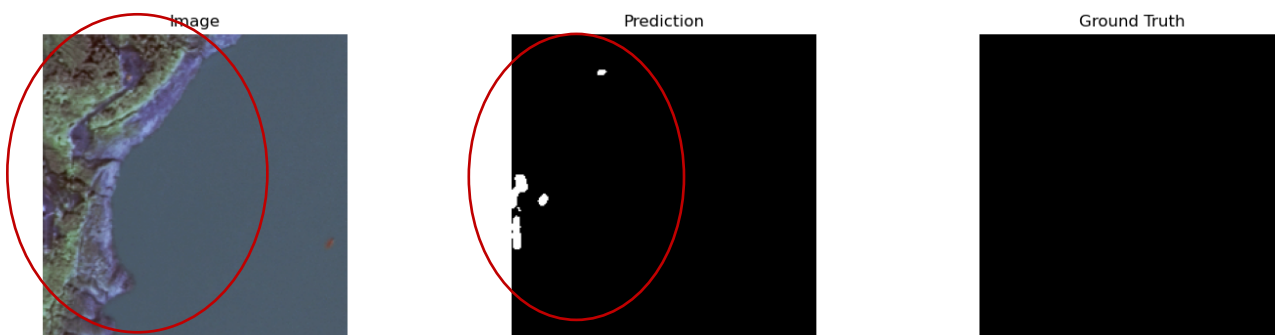


Best Case 58 : Chongqing - Dice: 1.0000, HD95: 0.0000



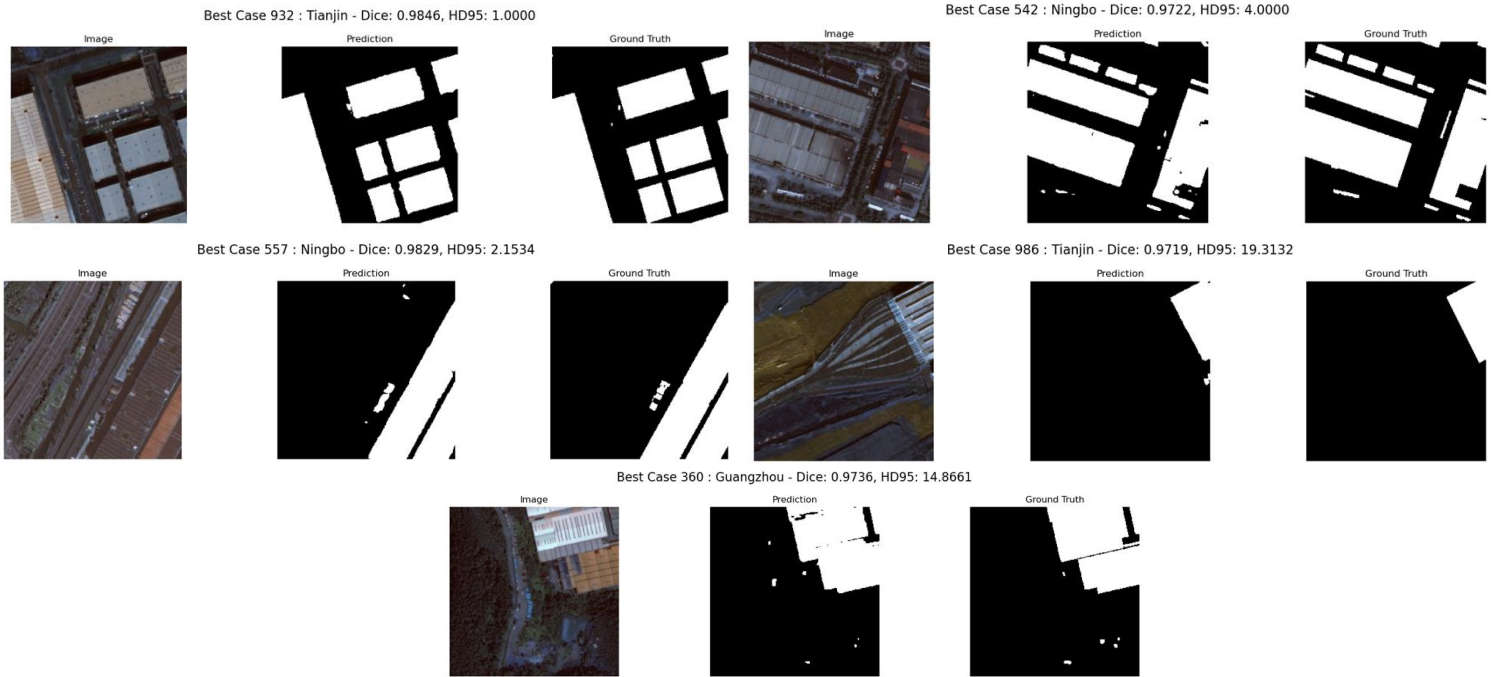
Overall looking at the cases where the Dice score is 0 we can see that they are all cases where the model is giving a false positive in images with no buildings and Dice Score 1 are cases where there aren't any buildings and the model is able to predict that. For the most part we can see that most of the cases where the model is making false positives are cases where we are dealing with more remote natural settings such as the shores of bodies of water or forested areas. Or they are cases where they only have infrastructure such as Roads or Highways. However, in many ways it is difficult to find an overall pattern distinguishing between cases where the Dice score is 0 and the model makes a false positive versus cases where the Dice Score is 1 and the model gets it right. Take for example Case 58 and Case 53 they are practically the same image to the naked eye yet in case 53 the model makes a false positive whereas in case 58 the model gets it right. It appears that some of the white highlights on the shore from what looks like exposed rocks might be confusing the model.

Worst Case 53 : Chongqing - Dice: 0.0000, HD95: 316.7838



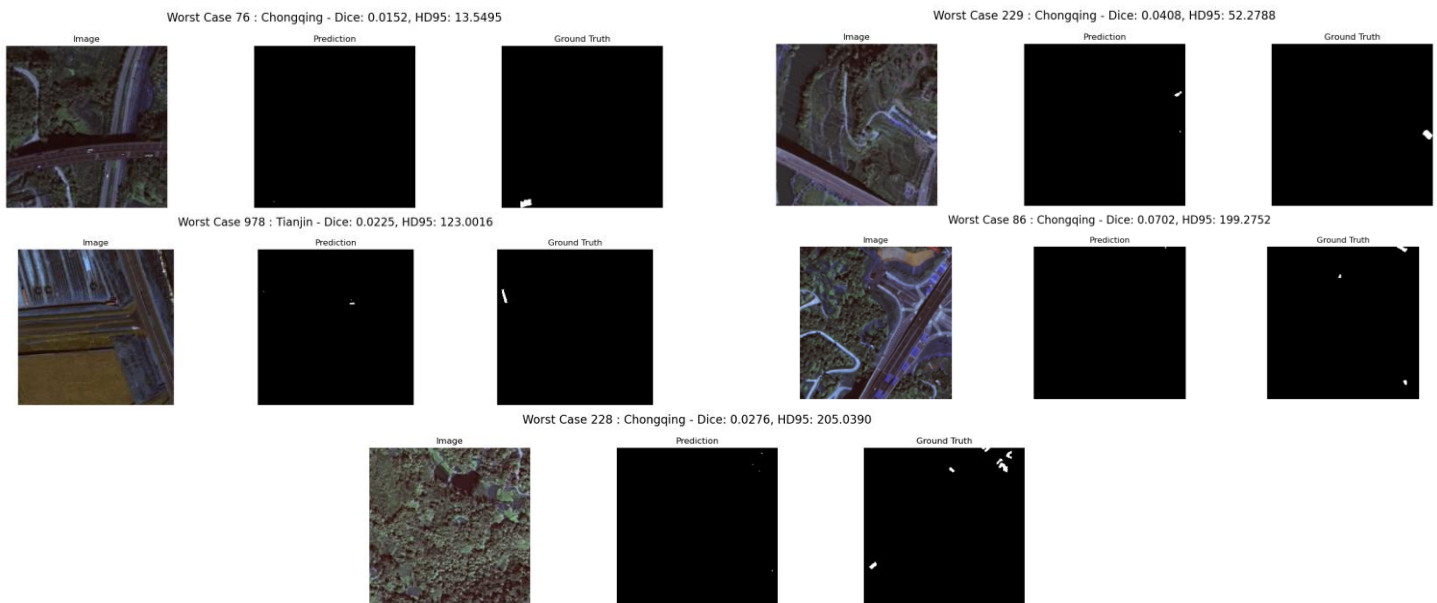
Similarly, when it comes to images of infrastructure like highways as illustrated in case 59 and 75 it occasionally completely fails to detect pretty obvious buildings. In these cases it appears the global attention mechanism might actually be confusing the model in these cases; it could be possible that when the model sees a highway it ignores other details. This would explain why cases 6, 13, and 23 it correctly predicts there aren't any buildings in the images.

TOP 5 BEST CASES BY DICE SCORE (IGNORING 0 & 1 DICE SCORES)



Looking at our Top 5 Cases we can see what we have discussed so far. Overall Performance is strong in moderately Dense Urban Areas especially in cases where we have simple shapes with large buildings like warehouses with distinctly colored roofs and cases where we have regular grid like patterns like in the case 932. Its strong performance with urban areas with regular grids might be a benefit of the Transformer architectures global self-attention capturing this global pattern more easily. While not clearly present in these 5 cases, but as we saw earlier, the model struggles capture some of the more fine-grained details and creating more fine grained and sharp segmentation decision boundaries, we can see this even in case 932 and 360 we despite it being a pretty straightforward shape the boundaries are bit fuzzy.

BOTTOM 5 CASES BY DICE SCORE (IGNORING 0 & 1 DICE SCORES)

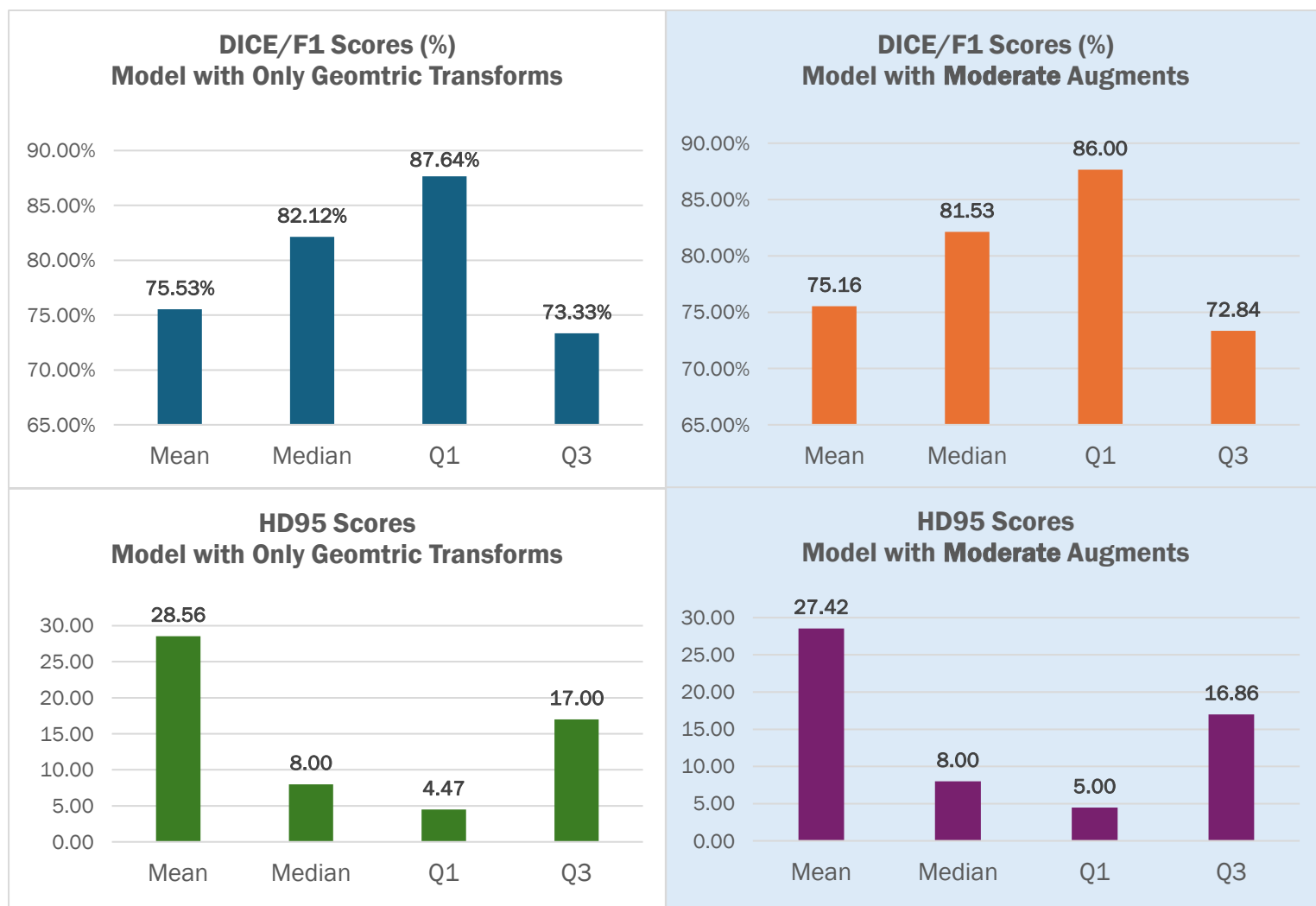


Our worst cases show a similar result to what we discussed earlier when discussing the cases where the Dice Score was 0. Cases with more natural settings like heavily forested areas, regions with few or no buildings, and images with large highways the model overall struggle to identify buildings and generate accurate segmentation masks.

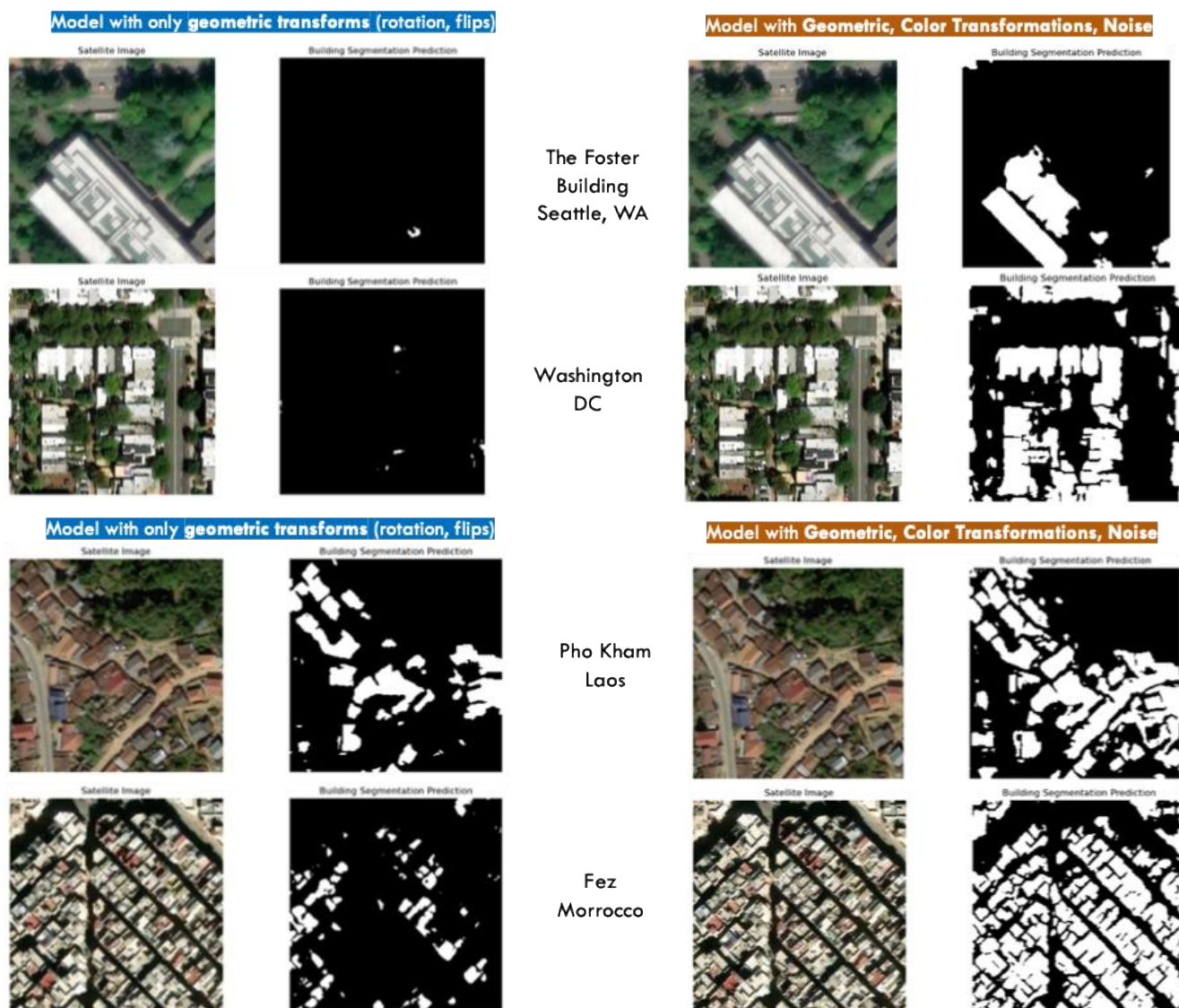
Experiments Comparing Effect of different Levels of Augmentation on Overall Generalizability of Our Model

As mentioned earlier when training our model, we experimented on the effect of various different levels of augmentation on the performance of our model. One of things we were curious about was the effect of only applying only Geometric Transformation (like was done in the original TransUNet model) and compare that against a more robust moderate augmentation pipeline that included Geometric, Color transformations alongside Noise on the overall performance and generalizability of our model.

Overall, in regard to the difference in performance on our test set we can see that for the most part the results are fairly similar, almost identical. For example the model with only Geometric Transforms had a 0.37 percentage point higher mean Dice Score compared to the Model with Moderate Augments while for the HD95 Score the Model with the Moderate Augments had a 1.14 pixel lower HD95 Score (better) than the Model with only Geometric Transforms. However, we really are splitting hairs here, the difference is small enough to be simple a product of random chance. There really is no practical difference



To test difference in generalizability between the Model trained with only Geometric Augments against the model trained with the Moderate Augments, as well as our overall generalizability of our model, we tested our model on Satellite Images from ArcGIS World Satellite Imagery for various cities around the world. You can see some of our results below



Overall looking at these results we can start to see the benefits of training with more moderate data augments as it has resulted in a far more robust and more generalizable model. In each of these cases we can see that the model with the more moderate augments far outperforms the model with only the geometric transforms/augments. In each of these cases the model with the moderate augments is able to pick up far more detail and far more buildings from the satellite images. In the image from Washington DC for example the model with only Geometric Augments completely failed at picking out any of the buildings while the model with the more moderate augments was able to segment most if not all.

In the case of Fez, Morocco we can see a similar case, the model limited to geometric transforms only identifies fragmented, incomplete patches of buildings. However, the case of Fez again also highlights the

benefit of Transformer architecture because the model with Moderate augments performed excellently in this sort of structure urban environment with grids, however it still struggles at capturing the fine details and creating sharp segmentation boundaries. The fact is in a dense, complex urban environment like Fez, buildings are not isolated objects but part of a large, continuous fabric. The Transformer's self-attention mechanism allows the model to process the image holistically, recognizing that disparate patches of rooftops are contextually related and part of the same interconnected structure. This ability to model long-range dependencies also the model to accurately segment such complex scenes, preventing fragmented results.

Looking at the case of the Foster Building at UW, even when we have large, distinct buildings the model limited to geometric transforms still struggles identifying only small, incorrect artifacts. The more robust modestly augment trained model, however, correctly outlines partially the general footprint of the building.

Finally the Image of Pho Kham, a village in Loas, highlights that overall, both the model with only the geometric transforms/augments and the model with more moderates transforms perform remarkably well even in more rural settings, such in the case of this rural village. The Model with Moderate Augments still performs better and is able to segment a greater number of buildings, but the performance of the model limited to only geometric transforms is till relatively decent

Overall the conclusion is clear: training the model with a moderate data augmentation pipeline, including changes in color and the addition of noise, forces it to learn the fundamental and invariant features of buildings, such as shape and texture. This approach prevents the model from overfitting to superficial characteristics like the specific colors or lighting conditions present in the training set.

At the same time we can also see it leads to a more robust and reliable model that demonstrates strong generalizability. This robustness is proven by its consistent performance across diverse geographic regions around the world from various different countries and completely different environments. Alongside the it also does decent job adapting to more noisy and less clean imagery captured by different satellites not just the GoaFen-7 Satellite and the clean images in the original GF7 dataset that it was trained on.

Discussion of Findings and Insights

The model demonstrates strong performance in dense urban areas but struggles with false positives in regions with few or no buildings—often identifying structures where none exist. This disparity may stem from our hybrid architecture, in which the transformer's global attention mechanism captures long-range dependencies and contextual patterns more effectively. In urban settings, where buildings tend to follow consistent grids and repeating patterns, this capability appears to enhance accuracy. However, the model's advantage diminishes in sparsely populated rural areas, where such structural regularities are less prevalent, leading to reduced precision and higher false detection rates.

We also compared the mean Dice coefficient and mean HD95 across the six major cities. The performance remained consistent across locations, with no significant differences observed between cities.

Another key insight from our ViT+U-Net model experiments is that data augmentations require careful tuning. We observed that aggressive data augmentations often degraded model performance. For instance, color jittering—changing the random factors of color saturation, brightness and contrast in the image color space (Shijie et al., 2017)—is intended to increase variation in the training data. However, if not carefully tuned, these adjustments can introduce unrealistic color patterns that do not reflect actual satellite imagery, ultimately confusing the model rather than improving its robustness. This highlights the importance of calibrating augmentation intensity to preserve the fidelity of real-world visual characteristics without messing with the color signals too much. In contrast, simpler geometric transformations like horizontal and vertical flips or rotations improved performance on the test set. However, these improvements came at the cost of reduced generalizability when evaluating the model on unseen regions or datasets, suggesting that while basic augmentations help with overfitting, they may also narrow the model's adaptability to broader contexts.

Limitations and Future Work

The biggest limitation we have is the fact that the model was trained on a single limited dataset. Due to time and hardware constraints, our model was trained and evaluated on a single satellite imagery dataset (GaoFen-7), which is already extremely clean with images only from China thus acts to limit its generalizability across different geographies, sensor types, and architectural styles.

Training the U-Net model required significant computational resources, particularly due to its encoder-decoder architecture with skip connections, which increases both memory consumption and training time. Eventually, this limitation may hinder scalability and real-world deployment of the model.

Another limitation we recognize is the rural Underrepresentation. Since the dataset contains limited amounts of rural imagery (only 15% of the imagery is from rural areas), it is introducing potential bias and contributing to poor performance in low-density or building-scarce regions.

To enhance the performance and fairness of our building extraction model, future work should focus on expanding training data to include more diverse satellite imagery from a variety of geographic regions and landscapes. This includes rural and underrepresented areas, leveraging datasets such as the Rural Home Annotation Dataset and SpaceNet 2 to ensure better generalizability across both urban and non-urban contexts. Additionally, experimenting with alternative transformer-based architectures—such as MobileViT for lightweight and mobile-friendly deployment—may yield a better tradeoff between performance and

computational cost. Or Transformers Like SWIN, to See if it improves Performance. According to Han et al. (2024), a Swin-UNet performed really well for the GF7 dataset. Conducting systematic ablation studies could also help us quantify the individual contributions of different model components, including encoder types, skip connections, and loss functions, allowing for more targeted architectural refinement. Since our current model is primarily a proof of concept, it offers substantial opportunities for optimization, including refining parameters to enhance performance, and there is a lot of areas where we can trim weight. Lastly, it is important to integrate fairness assessments into our evaluation pipeline by introducing metrics that measure disparities in performance across different demographic, geographic, and infrastructural styles. We plan to implement equalized odds fairness metric to reduce the quality-of-service harms. This will help ensure the model performs reliably and equitably in real-world applications.

Ethical Considerations

Prioritizing ethical practices, including inclusivity, fairness, and transparency in AI algorithms, can build trust among stakeholders and the wider community, fostering responsible innovation (Rane, 2023). Therefore, ethical guidelines for our future model deployment are essential. The biggest ethical concern lies in the underrepresentation of rural areas. Compared to urban populations, rural communities may suffer disproportionately from quality-of-service harms—such as false positives in building detection—that can misinform planning, resource allocation, or disaster response efforts. If the model falsely identifies a building in a rural area, it could lead to misguided infrastructure development or inefficient emergency interventions. This amplifies existing infrastructural inequities and risks further marginalizing already underserved populations. To mitigate these inequalities, one possible way would be to weigh rural cases more heavily to counteract the fact there are fewer rural cases in the dataset. And we also could include datasets with more rural satellite images.

Conclusion

Our Hybrid U-Net model prototype demonstrated competitive performance in building segmentation, especially well in dense urban areas. While it achieved strong accuracy overall, it faced challenges with false positives in regions with few or no buildings. We found that careful use of data augmentations significantly boosted both model performance and generalizability. Additionally, using advanced optimizer and scheduler configurations—specifically AdamW with AMSGrad and Cosine Annealing Warm Restarts—greatly improved training stability and convergence speed. Overall, our findings highlight the potential of hybrid architectures for scalable and fair building extraction, particularly in enhancing mapping efforts across diverse and underrepresented geographic regions.

Works Cited

- Bazi, Y., Bashmal, L., Rahhal, M. M. A., Dayil, R. A., & Ajlan, N. A. (2021). Vision Transformers for Remote Sensing Image Classification. *Remote Sensing*, 13(3), 516. <https://doi.org/10.3390/rs13030516>
- Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. L., & Zhou, Y. (2021). *TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation* (No. arXiv:2102.04306). arXiv. <https://doi.org/10.48550/arXiv.2102.04306>
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.
- Gugger, S., & Howard, J. (2018, July 2). *AdamW and Super-convergence is now the fastest way to train neural nets*. Fast.AI. <https://www.fast.ai/posts/2018-07-02-adam-weight-decay.html>
- Han, R., Fan, X., & Liu, J. (2024). EUNet: Edge-UNet for Accurate Building Extraction and Edge Emphasis in Gaofen-7 Images. *Remote Sensing*, 16(13), 2397.
- Ji, Z., Sun, H., Yuan, N., Zhang, H., Sheng, J., Zhang, X., & Ganchev, I. (2024). BGRD-TransUNet: A Novel TransUNet-Based Model for Ultrasound Breast Lesion Segmentation. *IEEE Access*, 12, 31182–31196. <https://doi.org/10.1109/ACCESS.2024.3368170>
- Zhang, X., & Tan, Y. (n.d.). *Transunet*. Retrieved June 9, 2025, from <https://tianjinteda.github.io/Transunet.html>
- Shijie, J., Ping, W., Peiyi, J., & Siping, H. (2017, October). Research on data augmentation for image classification based on convolution neural networks. In 2017 Chinese automation congress (CAC) (pp. 4165-4170). IEEE.
- Rane, N. (2023). Transformers in Intelligent Architecture, Engineering, and Construction (AEC) Industry: Applications, Challenges, and Future Scope. Engineering, and Construction (AEC) Industry: Applications, Challenges, and Future Scope (October 24, 2023).