AI Roof Visualization Tool For Construction

Introduction:

This project focuses on developing an advanced AI-driven roof visualization tool designed to assist homeowners and contractors in selecting the right roofing materials with confidence. By leveraging cutting-edge computer vision and image processing systems, the tool enables users to virtually preview different shingle options on their roofs, ensuring a seamless assessment of architectural compatibility and visual appeal before making a purchase.

The application accurately overlays the user-selected shingle samples onto user-uploaded images, delivering photorealistic previews in seconds. This real-time visualization capability simplifies the exploration of different roofing materials, providing an interactive and immersive experience for users.

The solution was specifically designed for businesses in the roofing and home improvement industry, to streamline the decision-making process while cutting operational costs. The objective of this project was to develop a pipeline capable of accurately detecting the roof structure and overlaying the shingle textures while preserving the orientation and spatial consistency.

Client Details:

Name: Confidential | **Industry**: Construction, Software | **Location**: USA

Technologies:

Computer Vision, Instance Segmentation, Object Detection, Image Processing, OpenCV, NumPY, PyTorch, CUDA, SAM2, Grounding DINO, FastAPI, AWS, Make Sense, YOLO, Mask-R-CNN, Detectron2.

AI Roof Visualization Tool For Construction

Project Description:

The roof visualization pipeline was built by integrating the two problem statements below.

1. Detecting the Roof Area

Developing a roof prediction system required training multiple instance segmentation models, including YOLO and Mask R-CNN variants with different ResNet backbones. These state-of-the-art computer vision models leverage Convolutional Neural Networks (CNNs) to detect and segment objects in images accurately. While YOLO prioritizes speed and real-time detection, Mask R-CNN excels at achieving fine-grained object localization and pixel-level segmentation.

Training these models for instance segmentation required a large dataset of high-resolution house images with corresponding roof annotations. To ensure high evaluation accuracy, it was important to annotate different parts of the image, such as trees, windows, and other obstructions that interfere with the roof area. Proper logging and validation of segmentation results ensured the robustness of the system across different architectural styles.

For instant mask prediction in cases where image datasets were unavailable for training, Grounding DINO and SAM2 models were used to predict roof segments. These models enabled efficient segmentation even in the absence of a dedicated training dataset and compute resources, ensuring reliable performance in real-world scenarios.

To enhance user interaction as per client's requests, we also implemented an interface feature that allows users to manually adjust the mask areas as needed, providing greater flexibility and control over the segmentation results.

2. Warping the Shingles

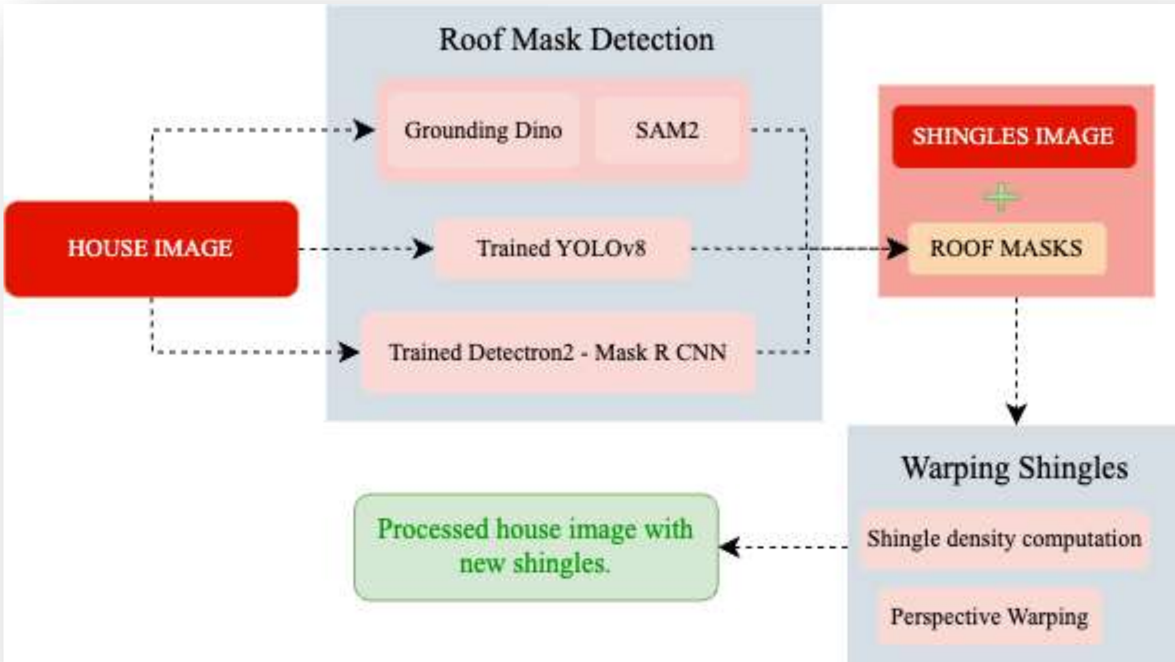
Applying selected shingles to the predicted roof segment presented several challenges. The selected shingle is a small sample of the material to be applied, requiring precise extrapolation over the roof area. Key challenges included:

1. **Density Estimation:** Determining the required repetitions of individual tiled shingles.
2. **Perspective Adjustments:** The orientation of the roof structure varies based on the camera angle and architectural design.

To address these challenges, an algorithm was developed to accurately warp the extrapolated shingle image onto the roof using perspective warping techniques. This process ensured realistic and seamless application, aligning the shingle pattern with the roof's structural geometry.

AI Roof Visualization Tool For Construction

Pipeline:



Variables impacting mask detection and warping:

1. Annotations prepared for images
2. Image resolution used for training the model
3. Batch size used for training
4. Complete size of the training dataset
5. GPU type and CUDA cores
6. GPU Memory
7. Prompt used in Grounding DINO
8. Images clicked from different points of view and camera angle
9. Masks with more than 4 coordinates.

Key measures to overcome optimization challenges:

1. Use of dedicated EC2 instances for model training and inference.
2. Algorithm for finding approximate bounding quadrilateral thereby improving warping of irregular mask area.