

UAV-based Visual Remote Sensing for Automated Building Inspection

Kushagra Srivastava^{1*}, Dhruv Patel^{1*}, Aditya Kumar Jha², Mohhit Kumar Jha², Jaskirat Singh³, Ravi Kiran Sarvadevabhatla¹, Pradeep Kumar Ramancharla¹, Harikumar Kandath¹, and K. Madhava Krishna¹

¹ International Institute of Information Technology, Hyderabad, India
{kushagra2000, dhruv.r.patel114}@gmail.com

{ravi.kiran, harikumar.k, ramancharla, mkrishna}@iiit.ac.in

² Indian Institute of Technology, Kharagpur, India

{aditya.jha, mohhit.kumar.jha2002}@kgpian.iitkgp.ac.in

³ University of Petroleum and Energy Studies, Dehradun, India
juskirat2000@gmail.com

Abstract. Unmanned Aerial Vehicle (UAV) based remote sensing system incorporated with computer vision has demonstrated potential for assisting building construction and in disaster management like damage assessment during earthquakes. The vulnerability of a building to earthquake can be assessed through inspection that takes into account the expected damage progression of the associated component and the component's contribution to structural system performance. Most of these inspections are done manually, leading to high utilization of manpower, time, and cost. This paper proposes a methodology to automate these inspections through UAV-based image data collection and a software library for post-processing that helps in estimating the seismic structural parameters. The key parameters considered here are the distances between adjacent buildings, building plan-shape, building plan area, objects on the rooftop and rooftop layout. The accuracy of the proposed methodology in estimating the above-mentioned parameters is verified through field measurements taken using a distance measuring sensor and also from the data obtained through Google Earth. Additional details and code can be accessed from <https://uvrsabi.github.io/>

Keywords: Building Inspection, UAV-based Remote Sensing, Segmentation, Image Stitching, 3D Reconstruction.

1 Introduction

Traditional techniques to analyze and assess the condition and geometric aspects of buildings and other civil structures involve physical inspection by civil experts according to pre-defined procedures. Such inspections can be costly, risky, time-consuming, labour and resource intensive. A considerable amount of research

* denotes equal contribution

has been dedicated to automating and improving civil inspection and monitoring through computer vision. This results in less human intervention and lower cost while ensuring effective data collection. Unmanned Aerial Vehicles (UAVs) mounted with cameras have the potential for contactless, rapid and automated inspection and monitoring of civil structures as well as remote data acquisition.

Computer vision-aided civil inspection has two prominent areas of application: damage detection and structural component recognition [1]. Studies focused on damage detection have used heuristic feature extraction methods to detect concrete cracks [2, 3, 4], concrete spalling [5, 6], fatigue cracks [7, 8] and corrosion in steel [9, 10, 11]. However, heuristic-based methods do not account for the information that is available in regions around the defect and have been replaced with deep learning-based methods. Image classification [12, 13, 14], object detection [15, 16], semantic segmentation [17, 18, 19] based methods have been used to successfully detect and classify the damage type. On the contrary, structural component analysis involves detecting, classifying and studying the characteristics of a physical structure. Hand-crafted filters [20, 21], point cloud-based [22, 23, 24, 25], and deep learning-based [26, 27, 28, 29] methods have been used to assess structural components like columns, planar walls, floor, bridges, beams and slabs. There also has been an emphasis on developing architectures for Building Information Modelling (BIM) [30, 31, 32] that involves analysis of physical features of a building using high resolution 3D reconstruction.

Apart from structural component recognition, it is also essential to assess the risk posed by earthquakes to buildings and other structural components. This is a crucial aspect of inspection in seismically active zones. Accurate seismic risk modeling requires knowledge of key structural characteristics of buildings. Learning-based models in conjunction with street imagery [33, 34] have been used to perform building risk assessments. However, UAVs can also be used to obtain information in areas difficult to access by taking a large number of images and videos from several points and different angles of view. Thus, UAVs demonstrate huge potential when it comes to remote data acquisition for pre- and/or post-earthquake risk assessments [35].

The main contributions of this paper are given below.

1. Primarily, we automate the inspection of buildings through UAV-based image data collection and a post-processing module to infer and quantify the details. This in effect avoids manual inspection, reducing the time and cost.
2. We estimate the distance between adjacent buildings and structures. To the best of our knowledge, there has not been any work that has addressed this problem.
3. We develop an architecture that can be used to segment roof tops in case of both orthogonal and non-orthogonal view using a state-of-the-art semantic segmentation model.
4. The software library for post-processing collates different algorithms used in computer vision along with UAV state information to yield an accurate estimation of the distances between adjacent buildings, building plan-shape, building plan area, objects on the rooftop, and rooftop layout. These pa-

rameters are key for the preparation of safety index assessment for buildings against earthquakes.

2 Related Works

2.1 Distance between Adjacent Structures

The collision between adjacent buildings or among parts of the same building during strong earthquake vibrations is called pounding [36]. Pounding occurs due to insufficient physical separation between adjacent structures and their out-of-phase vibrations resulting in non-synchronized vibration amplitudes. Pounding can lead to the generation of a high-impact force that may cause either architectural or structural damage. Some reported cases of pounding include i) The earthquake of 1985 in Mexico City [37] that left more than 20% of buildings damaged, ii) Loma Prieta earthquake of 1989 [38] that affected over 200 structures, iii) Chi-Chi earthquake of 1999 [39] in central Taiwan, and iv) Sikkim earthquake (2006) [40]. Methods such as Rapid Visual Screening (RVS), seismic risk indexes, and vulnerability assessments have been developed to analyze the level of damage to a building [41]. In particular, RVS-based methods have been used for pre-and/or post-earthquake screening of buildings in earthquake-prone areas. The pounding effect is considered as a vulnerability factor by RVS methods like FEMA P-154, FEMA 310, EMS-98 Scale, NZSEE, OSAP, NRCC, IITK-GSDMA, EMPI and RBTE-2019 [42].

The authors in [43] present a UAV-based site survey using both Nadir and Oblique images for appropriate 3D modelling. The integration of nadir UAV images with oblique images ensures a better inclusion of facades and footprints of the buildings. Distances between the buildings in the site were manually measured from the generated dense point cloud. We use the 3D reconstruction of the structures from images in conjunction with conditional plane fitting for estimating the distance between adjacent structures.

2.2 Plan Shape and Roof Area Estimation

The relationship between the center of stiffness and gravity's eccentricity is influenced by shape irregularities, asymmetries, or concavities, as well as by building mass distributions. For any structure, if the centre of stiffness is moved away from the centre of gravity during ground motion, more torsion forces are produced [44]. When a building is shaken by seismic activity, this eccentricity causes structures to exhibit improper dynamic characteristics. Hence, the behavior of a building under seismic activity also depends on its 3D configuration, plan shape and mass distribution [45]. *Plan shape and Roof Area* is needed for calculating the Floor Space Index (FSI). FSI is the ratio of the total built-up area of all the floors to the plot area. FSI is a contributing factor in assessing the extent of the damage and is usually fixed by the expert committee.

Roof-top segmentation has been considered as a special case of 3D plane segmentation from point clouds and can be achieved through model fitting [46],

region growing [47], feature clustering [48] and global energy optimization-based methods [49]. Studies focused on these methods have been tested on datasets where the roof was visible orthogonally through satellite imagery [50] and LiDAR point clouds [46, 47, 48]. The accuracy of these methods depends on how the roof is viewed. In case of a non-orthogonal view, these methods must be used in conjunction with some constraints. On the contrary, learning-based methods [51] have been developed that specifically segment out roofs. The neural networks employed in these methods have been trained on satellite imagery and do not perform well in non-orthogonal roof-view scenarios. Our approach is to segment out roofs when viewed both orthogonally and non-orthogonally by training a state-of-the-art semantic segmentation model on a custom roof-top dataset.

2.3 Roof Layout Estimation

Roof Layout Estimation refers to identifying and locating objects present on the roof such as air conditioner units, solar panels, etc. Such objects are usually non-structural elements (NSE). As the mass of the NSE increases, the earthquake response of the NSE starts affecting the whole building. Hence, they need to be taken into account for design calculations. Furthermore, the abundance of these hazardous objects may create instabilities on the roof making it prone to damage during earthquakes. Estimating the *Roof Layout* is not as trivial as in the case of satellite images, since the UAV has altitude limitations along with camera Field of View (FOV) constraints, thereby limiting us from obtaining a complete view of the roof in a single image. Moreover, we cannot rely on satellite images because it does not provide us with real time observation of our location of interest. Hence, we solve this problem by first stitching a large number of images with partially visible roofs to create a panoramic view of the roof and then we apply object detection and semantic segmentation to get the object and roof masks respectively.

Various techniques for image stitching can be roughly distinguished into three categories: direct technique [52, 53, 54], feature-based technique [55, 56, 57] and position-based technique [58]. The first category performs pixel-based image stitching by minimizing the sum of the absolute difference between overlapping pixels. These methods are scale and rotation variant and to tackle this problem, the second category focuses on extracting a set of images and matching them using feature based algorithms which includes SIFT, SURF, Harris Corner Detection. These methods are computationally expensive and fail in the absence of distinct features. The third category stitches images sampled from videos through their overlapping FOV. Due to the inability to obtain accurate camera poses, not much research has been conducted on this approach. In this paper, we present an efficient and reliable approach to make use of the camera poses and stitch a large set of images avoiding the problems of image drift and expensive computation associated with the first two categories.

3 Data Collection

This section discusses the methods for gathering data that were utilized to carry out the research experiments in this study. DJI Mavic Mini⁴ UAV is used for gathering visual data because of its high-quality image sensory system with an adjustable gimbal.

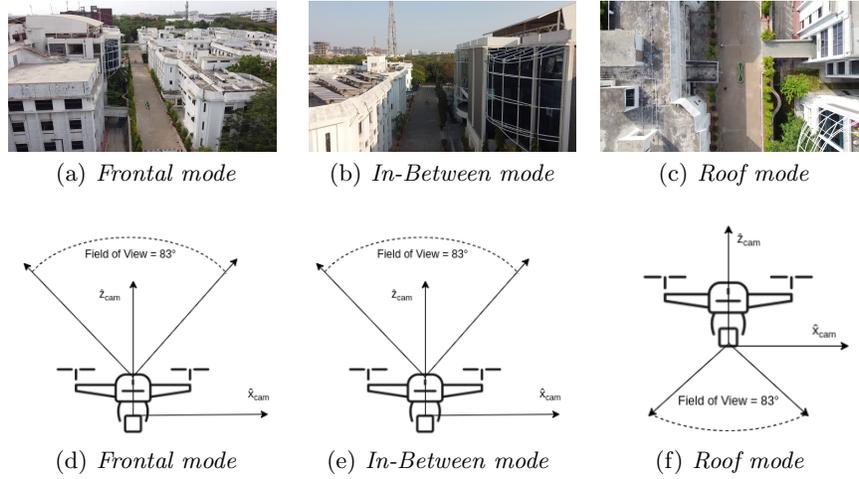


Fig. 1: Figures 1(a), 1(b), and 1(c) indicate the UAV's point of view while figures 1(d), 1(e), and 1(f) are representations of the respective coordinate system adopted.

For estimating the distance between adjacent structures, the images are collected in 3 different modes: *Frontal Mode*, *In-Between Mode* and *Roof Mode*. Fig. 1(a) shows the frontal face of the two adjacent buildings for which data was collected. In this mode, we focus on estimating the distance between the two buildings by analyzing only their frontal faces through a forward-facing camera. This view is particularly helpful when there are impediments between the subject buildings and flying a UAV between them is challenging. In fig. 1(b), the UAV was flown in-between the two buildings along a path parallel to the facade with a forward-facing camera. This mode enables the operators to calculate distances when buildings have irregular shapes. Lastly, for the *roof mode*, the UAV was flown at a fixed altitude with a downward-facing camera so as to capture the rooftops of the subject buildings. Fig. 1(c) is a pictorial representation of the *roof mode*. The *roof mode* helps in tackling occlusions due to vegetation and other physical structures.

⁴ UAV specification details can be found at the official DJI website: <https://www.dji.com/mavic-mini>

For *Rooftop Layout Estimation*, the UAV was flown at a constant height with a downward-facing camera, parallel to the plane of the roof. This helped in robust detection of NSE. To estimate the *Plan Shape and Roof Area*, a dataset comprising of around 350 images was prepared from the campus buildings and UrbanScene3D dataset [59]. The training set comprised of images scraped from the UrbanScene3D videos, *Buildings 4* and *6* and while the validation and test set comprised of the *Buildings 3, 5* and *7*. This was done to ensure that the model learns the characteristic features of a roof irrespective of the building plan shape. Out of these, 50 images had fully-visible buildings while the rest contained partially-visible buildings.

4 Methodology

We propose different methods to calculate the distance between the adjacent buildings using plane segmentation; estimate the roof layout using Object Detection and large scale image stitching; estimate the roof area and plan shape using roof segmentation as shown in Fig. 2.

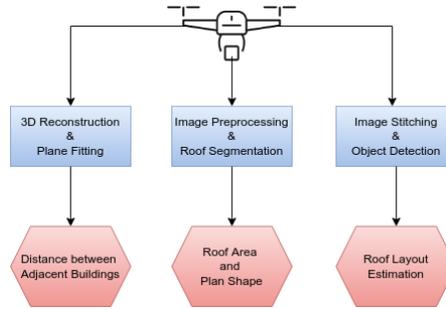


Fig. 2: Architecture of automated building inspection using the aerial images captured using UAV. The odometry information of UAV is also used for the quantification of different parameters involved in the inspection.

4.1 Distance Between Adjacent Buildings

We use plane segmentation to obtain the distance between the two adjacent buildings. We have divided our approach into three stages as presented in Fig. 3. In *Stage I*, images were sampled from the video captured by the UAV and panoptic segmentation was performed using a state-of-the-art network [60], to obtain vegetation-free masks. This removes trees and vegetation near the vicinity of the buildings and thus improves the accuracy of our module. Fig. 4 shows the impact of panoptic segmentation for *frontal mode*. In *Stage II*, the masked images were generated from the binary masks and the corresponding images. The

masked images are inputs to a state-of-the-art image-based 3D reconstruction library [61, 62] which outputs a dense 3D point cloud and the camera poses through Structure-from-Motion. Our approach for all the three modes is same for the first two stages.

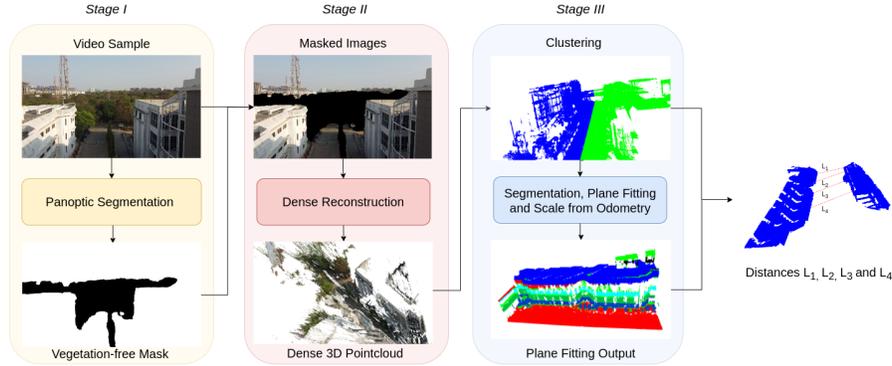


Fig. 3: Architecture for estimation of distance between adjacent structures.

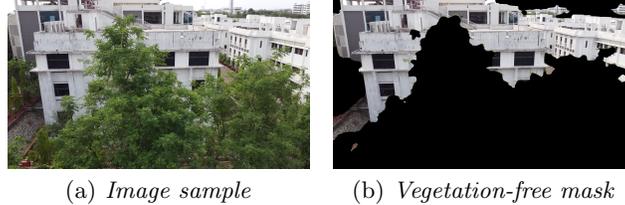


Fig. 4: Removal of vegetation from the sample images enhances the structural features of the buildings in the reconstructed 3D model leading to more accurate results.

In *Stage III*, we aim to extract planes from the given point cloud that are essential to identify structures such as roof and walls of the building. We employ the co-ordinate system depicted in Fig. 1. We can divide this task into two parts: i) Isolation of different building clusters and ii) Finding planes in each cluster. Isolation of the concerned buildings is done using euclidean clustering thereby creating two clusters. For instance, in *Roof Mode* the clusters are distributed on either side of the Y-axis. Similarly, for *In-Between* and *Frontal mode* the clustering happens about the Z-axis. In order to extract the planes of interest, we slice each cluster along a direction parallel to our plane of interest, into small segments of 3D points. For instance, in *Roof mode* we are interested in fitting

a plane along the roof of a building; therefore, we slice the building perpendicular to ground normal, i.e, the Z-axis. Finally, Random Sample Consensus (RANSAC) algorithm is applied for each segment of 3D points to iteratively fit a plane and obtain a set of parallel planes as shown under the *Stage III* in Fig. 3.

Our approach selects a plane from the set of planes estimated in *Stage III* for each building based on the highest number of inliers. As stated above, for each mode, the selected planes for the adjacent buildings have the same normal unit vector. Further, we sample points on these planes to calculate the distance between the adjacent buildings at different locations. The scale estimation is done by using the odometry data received from the UAV and the estimated distance is scaled up to obtain the actual distance between the adjacent buildings. This was done by time-syncing the flight logs, that contains GPS, Barometer and IMU readings, with the sampled images.

4.2 Plan Shape and Roof Area Estimation

The dataset for roof-top of various buildings was collected as described in Section 3. This dataset was used to estimate the layout and area of the roof through semantic segmentation. The complete *Plan Shape* module has been summarized in Fig. 5. For the task of roof segmentation, we use a state-of-the-art semantic segmentation model, LEDNet [63]. The asymmetrical architecture of this network leads to reduction in network parameters resulting in a faster inference process. The split and shuffle operations in the residual layer enhances information sharing while the decoder’s attention mechanism reduces complexity of the whole network. We subject the input images to a pre-processing module that removes distortion from the wide-angle images. Histogram equalization is also performed to improve the contrast of the image. Data augmentation techniques

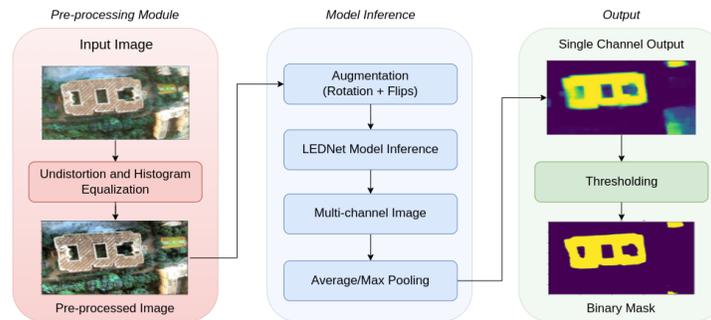


Fig. 5: Architecture of the *Plan Shape* module providing the segmented mask of the roof as output from the raw input image.

(4 rotations of 90° + horizontal flip + vertical flip) were used during inference to

improve the network’s performance and increase robustness. The single-channel grey-scale output is finally thresholded to obtain a binary mask. The roof area from the segmentation masks can be obtained by using Equation 1 where C is the contour area (in $pixels^2$), obtained from the segmented mask, D is the depth of the roof from the camera (in m) and f is the focal length of the camera (in $pixels$) used.

$$Area(m^2) = C \times (D/f)^2 \quad (1)$$

4.3 Roof Layout Estimation

The data for this module was collected as described in Section 3. Due to the camera FOV limitations and to maintain good resolution, it is not possible to capture the complete view of the roof in a single image, especially in the case of large sized buildings. Hence, we perform large scale stitching of partially visible roofs followed by NSE detection and roof segmentation. Fig 6 shows the approach adopted for *Roof Layout Estimation*.

Large Scale Aerial Image Stitching: We exploit the planarity of the roof and the fact that the UAV is flown at a constant height from the roof. Instead of opting for homography, that relates two geometric views in case of image stitching, we opt for affine transformations. Affine transformations are linear mapping methods that preserve points, straight lines, and planes.

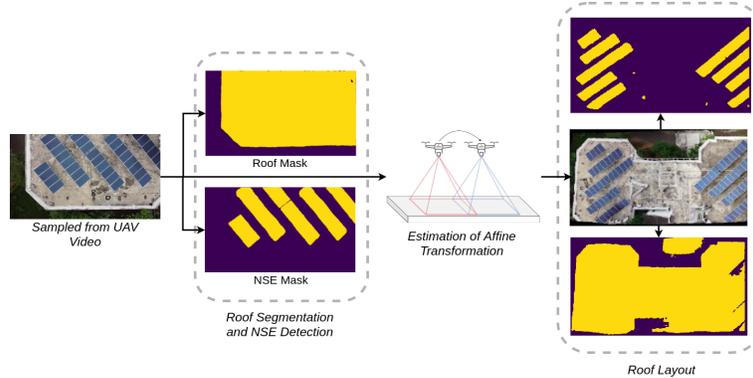


Fig. 6: We exploit the planarity of the roof and the fact that the distance between the UAV and the roof will be constant (the UAV is flown at a constant height). This enables us to relate two consecutive images through an affine transformation.

Let $I = \{i_1, i_2, i_3, \dots, i_N\}$ represent an ordered set of images sampled from a video collected as per Section 3. The image stitching algorithm implemented has been summarized below:

1. Features were extracted in image i_1 , using ORB feature detector and tracked in the next image, i_2 using optical flow. This helped in effective rejection of outliers.
2. The obtained set of feature matches across both the images were used to determine the affine transformation matrix using RANSAC.
3. Images i_1 and i_2 were then warped as per the transformation and stitched on a *canvas*.
4. Affine transformation was calculated between image i_3 and the previously warped image i_2 before it was stitched using steps 1 and 2. Image i_3 was then warped and stitched on the same *canvas*.
5. Step 4 was repeated for the next set of images, that is, affine transformation was calculated for image i_4 and the previously warped image i_3 before it was stitched.

Detecting Objects on Rooftop: For identification of NSE on the rooftop, we use a state-of-the art object detection model, Detic [64] because it is highly flexible and has been trained for large number of classes. In order to estimate the roof layout, it is essential to detect and locate the NSE as well as the roof from a query image. Note that we classify all the NSE as a single class. This information can then be represented as a semantic mask which will be to calculate the percentage of occupancy of the NSE. A custom vocabulary comprising of the NSE was passed to the model. The roof was segmented out using LEDNet as described in Section 4.2.

5 Results

This section presents the results for the different modules of automated building inspection using aerial images.

5.1 Distance Between Adjacent Buildings

We validated our algorithm on real aerial datasets of adjacent buildings and structures. In particular, we tested all the modes of this module on a set of adjacent buildings, *Buildings 1 and 2*, and also on *Building 3*, a U-shaped building. The resulting distances for all the modes can be visualized through Fig. 7. The corresponding distances visualized in Fig. 7 have been documented in Table 1 and 2. We obtain the ground truth from using a Time-of-Flight (ToF) based range measuring sensor⁵. This sensor has a maximum range of 60 meters. We also compare the results with that from *Google Earth*. It must be noted that using *Google Earth*, it is not possible to measure some distances due to lack of 3D imagery.

⁵ The ToF sensor can be found at: <https://www.terabee.com/shop/lidar-tof-range-finders/teraranger-evo-60m/>

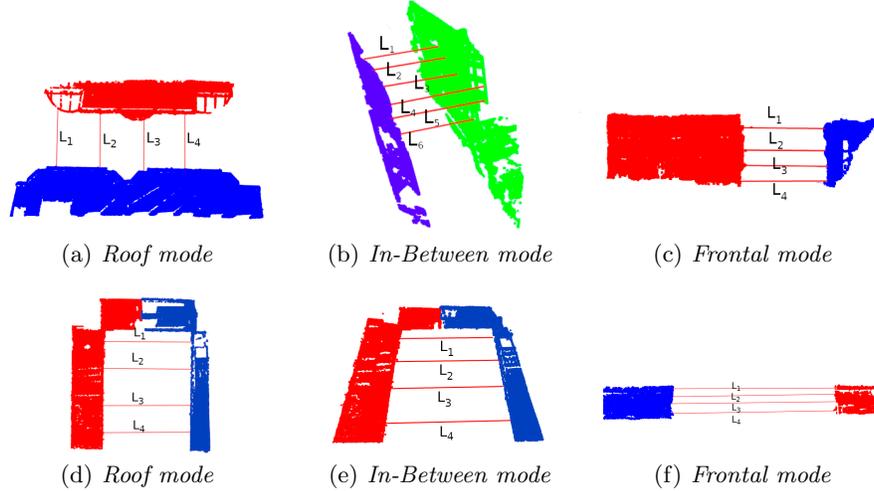


Fig. 7: 7(a), 7(b) and 7(c) and 7(d), 7(e) and 7(f) represent the implementation of plane fitting using piecewise-RANSAC in different views for *Buildings 1, 2* and *Building 3* respectively.

| Mode | <i>Buildings 1 and 2</i> | | | | | |
|-------------------|----------------------------|--------------|--------------|-----------|----------------------|-------------------|
| | Distance Reference | Ground Truth | Google Earth | Estimated | Error (Google Earth) | Error (Estimated) |
| <i>Roof</i> | L ₁ in Fig 7(a) | 16.40 m | 17.14 m | 16.70 m | 4.5% | 1.8% |
| | L ₂ in Fig 7(a) | 12.96 m | 12.91 m | 12.94 m | 0.3% | 0.15% |
| | L ₃ in Fig 7(a) | 12.01 m | 12.08 m | 11.97 m | 0.58% | 0.33% |
| | L ₄ in Fig 7(a) | 13.30 m | 13.00 m | 12.77 m | 2.31% | 3.98% |
| <i>In-Between</i> | L ₁ in Fig 7(b) | 13.31 m | 13.50 m | 13.22 m | 1.42% | 0.67% |
| | L ₂ in Fig 7(b) | 12.91 m | 12.40 m | 12.87 m | 3.95% | 0.31% |
| | L ₃ in Fig 7(b) | 12.30 m | 12.43 m | 12.12 m | 1.00% | 1.39% |
| | L ₄ in Fig 7(b) | 12.70 m | 12.87 m | 12.50 m | 1.33% | 1.57% |
| | L ₅ in Fig 7(b) | 13.95 m | 13.84 m | 13.87 m | 0.79% | 0.51% |
| | L ₆ in Fig 7(b) | 12.60 m | 12.69 m | 12.56 m | 0.71% | 0.31% |
| <i>Frontal</i> | L ₁ in Fig 7(c) | 16.96 m | 16.91 m | 16.92 m | 0.29% | 0.23% |
| | L ₂ in Fig 7(c) | 16.96 m | - | 16.78 m | - | 1.06% |
| | L ₃ in Fig 7(c) | 16.96 m | - | 17.13 m | - | 1.00% |
| | L ₄ in Fig 7(c) | 16.96 m | - | 17.05 m | - | 0.53% |

Table 1: Distances calculated for *Building 1 and 2* using our method and Google Earth for all three modes.

| Mode | <i>Building 3</i> | | | | | |
|-------------------|----------------------------|--------------|--------------|-----------|----------------------|-------------------|
| | Distance Reference | Ground Truth | Google Earth | Estimated | Error (Google Earth) | Error (Estimated) |
| <i>Roof</i> | L ₁ in Fig 7(d) | 33.28 m | 33.58 m | 33.26 m | 0.90 % | 0.06 % |
| | L ₂ in Fig 7(d) | 33.28 m | 33.63 m | 33.22 m | 1.05 % | 0.18 % |
| | L ₃ in Fig 7(d) | 33.28 m | 33.00 m | 33.28 m | 0.84 % | 0.00 % |
| | L ₄ in Fig 7(d) | 33.28 m | 33.35 m | 33.81 m | 0.21 % | 1.59 % |
| <i>In-Between</i> | L ₁ in Fig 7(e) | 33.28 m | 32.58 m | 33.11 m | 2.10 % | 0.51 % |
| | L ₂ in Fig 7(e) | 33.28 m | 33.12 m | 32.40 m | 0.48 % | 2.64 % |
| | L ₃ in Fig 7(e) | 33.28 m | 32.94 m | 32.78 m | 1.02 % | 1.50 % |
| | L ₄ in Fig 7(e) | 33.28 m | 32.25 m | 32.81 m | 3.09 % | 1.41 % |
| <i>Frontal</i> | L ₂ in Fig 7(f) | 33.28 m | 33.57 m | 33.26 m | 0.87 % | 0.06 % |
| | L ₁ in Fig 7(f) | 33.28 m | - | 33.60 m | - | 0.96 % |
| | L ₃ in Fig 7(f) | 33.28 m | - | 33.59 m | - | 0.93 % |
| | L ₄ in Fig 7(f) | 33.28 m | - | 33.99 m | - | 2.13 % |

Table 2: Distances calculated for *Building 3* using our method and Google Earth for all three modes.

5.2 Plan Shape and Roof Area Estimation

The roof area was estimated from images taken at different depths, that is, when the UAV was operated at different altitudes ranging from 50m to 100m. The module was tested on various campus buildings. The results in Table 3 were averaged out for all samples corresponding to the same building. The module estimates the roof area with an average difference of 4.7% with Google Earth data. Predicted roof masks of some buildings from LEDNet are shown in Fig 8.



Fig. 8: Roof Segmentation results for 4 buildings.



Fig. 9: We use LEDNet in both *Plan Shape and Roof Area Estimation* as well as *Roof Layout Estimation*. The trained model correctly segments out the roof in case of a non-orthogonal view, that is, when the downward-facing camera is not directly above the roof.

| Building | Area Measured using Google Earth | Estimated Area | Absolute Difference | Percentage Difference |
|-------------------|----------------------------------|------------------------|-----------------------|-----------------------|
| <i>Building 3</i> | 1859.77 m ² | 1939.84 m ² | 80.07 m ² | 4.3 % |
| <i>Building 4</i> | 350 m ² | 331.30 m ² | 18.70 m ² | 5.3 % |
| <i>Building 5</i> | 340 m ² | 329.55 m ² | 10.45 m ² | 3.1 % |
| <i>Building 6</i> | 3,127.60 m ² | 2936.82 m ² | 190.78 m ² | 6.1 % |

Table 3: Roof Area Estimation Results

5.3 Roof Layout Estimation

Data for *Roof Layout Estimation* was collected as described in Section 3. Images were sampled at a frequency of 10Hz. The video collected for *Roof Layout Estimation* was sampled at a frequency of 1 Hz generating 98 images. The results of image stitching can be visualized in Fig. 10(a) with the corresponding roof mask in Fig. 10(b) and the NSE mask in Fig. 10(c). The percentage occupancy was calculated by taking the ratio of object occupancy area ($pixels^2$) in Fig. 10(c) to total roof area ($pixels^2$) in Fig. 10(b). The final percentage occupancy obtained was 38.73%.

6 Discussion

We estimate the distance between adjacent structures using 3D reconstruction and conditional plane fitting and validate its performance on ground truth data from a ToF sensor. We also make a comparison of our proposed module with Google Earth and validate our superior performance. Moreover, it is not possible to employ Google Earth for this module universally due to the lack of 3D imagery for all the buildings. We validated our distance estimation algorithm and compared the results with the ground truth and Google Earth. We estimated the distance between adjacent structures with an average error of 0.94%, which is superior to Google Earth which performs with an average error of 1.36%. Our rooftop area estimation module performs with an average difference of 4.7%

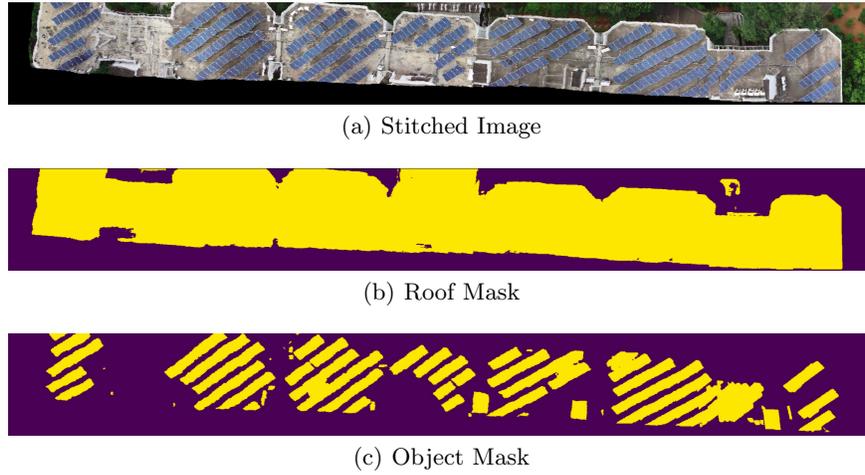


Fig. 10: Results for Roof Layout Estimation.

when compared to Google Earth. It is observed that the difference remains near-constant irrespective of the size of the rooftop area. Considering the irregular shape of the roofs, it is challenging to measure the ground truth of the roof area manually and it is also error-prone and resource-intensive. The roof layout was estimated through semantic segmentation, object detection, and large-scale image stitching of 98 images. We also detected NSE on the rooftops and found their percentage occupancy to be 38.73%.

7 Conclusion

This paper presented an implementation of a considerable amount of approaches that have been developed, aiming at modeling the structure of buildings. Seismic risk assessment of buildings involves the estimation of several structural parameters. It is important to estimate the parameters which can describe the geometry of buildings, plan shape of the rooftops, and size of the buildings. In particular, we estimated the distances between adjacent buildings and structures, plan shape of a building, roof area, and percentage of area occupied by NSE. We plan to release these modules in the form of an open-source library that can be easily used by non-computer vision experts. Future work includes quantifying the flatness of ground, crack detection, and identification of water tanks and staircase exits that could help in taking preliminary precautions for earthquakes.

Acknowledgement: The authors acknowledge the financial support provided by IHUB, IIT Hyderabad to carry out this research work under the project: IIT-H/IHub/Project/Mobility/2021-22/M2-003.

Bibliography

- [1] Billie F. Spencer, Vedhus Hoskere, and Yasutaka Narazaki. Advances in computer vision-based civil infrastructure inspection and monitoring. *Engineering*, 5(2):199–222, 2019. ISSN 2095-8099. <https://doi.org/https://doi.org/10.1016/j.eng.2018.11.030>. URL <https://www.sciencedirect.com/science/article/pii/S2095809918308130>. 2
- [2] Ikhlas Abdel-Qader, Osama Abudayyeh, and Michael Kelly. Analysis of edge-detection techniques for crack identification in bridges. *Journal of Computing in Civil Engineering - J COMPUT CIVIL ENG*, 17, 10 2003. [https://doi.org/10.1061/\(ASCE\)0887-3801\(2003\)17:4\(255\)](https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(255)). 2
- [3] Wenyu Zhang, Zhenjiang Zhang, Dapeng Qi, and Yun Liu. Automatic crack detection and classification method for subway tunnel safety monitoring. *Sensors*, 14(10):19307–19328, 2014. ISSN 1424-8220. <https://doi.org/10.3390/s141019307>. URL <https://www.mdpi.com/1424-8220/14/10/19307>. 2
- [4] Yu-Fei Liu, Soojin Cho, Billie Spencer, and Jian-Sheng Fan. Concrete crack assessment using digital image processing and 3d scene reconstruction. *Journal of Computing in Civil Engineering*, 30:04014124, 08 2014. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000446](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000446). 2
- [5] Ram Sebak Adhikari, Osama Moselhi, and Ashutosh Bagchi. A study of image-based element condition index for bridge inspection. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 30, page 1. IAARC Publications, 2013. 2
- [6] Stephanie Paal, Jong-Su Jeon, Ioannis Brilakis, and Reginald Desroches. Automated damage index estimation of reinforced concrete columns for post-earthquake evaluations. *Journal of Structural Engineering*, 141:04014228, 09 2015. [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0001200](https://doi.org/10.1061/(ASCE)ST.1943-541X.0001200). 2
- [7] Chul Min Yeum and Shirley J Dyke. Vision-based automated crack detection for bridge inspection. *Computer-Aided Civil and Infrastructure Engineering*, 30(10):759–770, 2015. 2
- [8] Mohammad R Jahanshahi, Fu-Chen Chen, Chris Joffe, and Sami F Masri. Vision-based quantitative assessment of microcracks on reactor internal components of nuclear power plants. *Structure and Infrastructure Engineering*, 13(8):1013–1026, 2017. 2
- [9] Hyojoo Son, Nahyae Hwang, Changmin Kim, and Changwan Kim. Rapid and automated determination of rusted surface areas of a steel bridge for robotic maintenance systems. *Automation in Construction*, 42:13–24, 2014. 2
- [10] Heng-Kuang Shen, Po-Han Chen, and Luh-Maan Chang. Automated steel bridge coating rust defect recognition method based on color and texture feature. *Automation in Construction*, 31:338–356, 2013. 2

- [11] Fátima NS Medeiros, Geraldo LB Ramalho, Mariana P Bento, and Luiz CL Medeiros. On the evaluation of texture and color features for nondestructive corrosion detection. *EURASIP Journal on Advances in Signal Processing*, 2010:1–7, 2010. [2](#)
- [12] Young-Jin Cha, Wooram Choi, and Oral Büyüköztürk. Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, 32(5):361–378, 2017. [2](#)
- [13] Lei Zhang, Fan Yang, Yimin Daniel Zhang, and Ying Julie Zhu. Road crack detection using deep convolutional neural network. In *2016 IEEE international conference on image processing (ICIP)*, pages 3708–3712. IEEE, 2016. [2](#)
- [14] Deegan J Atha and Mohammad R Jahanshahi. Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection. *Structural Health Monitoring*, 17(5):1110–1128, 2018. [2](#)
- [15] Chul Min Yeum, Shirley J Dyke, and Julio Ramirez. Visual data classification in post-event building reconnaissance. *Engineering Structures*, 155:16–24, 2018. [2](#)
- [16] Young-Jin Cha, Wooram Choi, Gahyun Suh, Sadegh Mahmoudkhani, and Oral Büyüköztürk. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civil and Infrastructure Engineering*, 33(9):731–747, 2018. [2](#)
- [17] Allen Zhang, Kelvin CP Wang, Baoxian Li, Enhui Yang, Xianxing Dai, Yi Peng, Yue Fei, Yang Liu, Joshua Q Li, and Cheng Chen. Automated pixel-level pavement crack detection on 3d asphalt surfaces using a deep-learning network. *Computer-Aided Civil and Infrastructure Engineering*, 32(10):805–819, 2017. [2](#)
- [18] Vedhus Hoskere, Yasutaka Narazaki, Tu Hoang, and BillieF Spencer Jr. Vision-based structural inspection using multiscale deep convolutional neural networks. *arXiv preprint arXiv:1805.01055*, 2018. [2](#)
- [19] Vedhus Hoskere, Yasutaka Narazaki, Tu A Hoang, and Billie F Spencer Jr. Towards automated post-earthquake inspections with deep learning-based condition-aware models. *arXiv preprint arXiv:1809.09195*, 2018. [2](#)
- [20] Zhenhua Zhu and Ioannis Brilakis. Concrete column recognition in images and videos. *Journal of computing in civil engineering*, 24(6):478–487, 2010. [2](#)
- [21] Christian Koch, S German Paal, Abbas Rashidi, Zhenhua Zhu, Markus König, and Ioannis Brilakis. Achievements and challenges in machine vision-based inspection of large concrete structures. *Advances in Structural Engineering*, 17(3):303–318, 2014. [2](#)
- [22] Xuehan Xiong, Antonio Adan, Burcu Akinci, and Daniel Huber. Automatic creation of semantically rich 3d building models from laser scanner data. *Automation in construction*, 31:325–337, 2013. [2](#)
- [23] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016. [2](#)

- [24] Mani Golparvar-Fard, Jeffrey Bohn, Jochen Teizer, Silvio Savarese, and Feniosky Peña-Mora. Evaluation of image-based modeling and laser scanning accuracy for emerging automated performance monitoring techniques. *Automation in construction*, 20(8):1143–1155, 2011. 2
- [25] Ruodan Lu, Ioannis Brilakis, and Campbell R Middleton. Detection of structural components in point clouds of existing rc bridges. *Computer-Aided Civil and Infrastructure Engineering*, 34(3):191–212, 2019. 2
- [26] Yuqing Gao and Khalid M Mosalam. Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, 33(9):748–768, 2018. 2
- [27] Xiao Liang. Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with bayesian optimization. *Computer-Aided Civil and Infrastructure Engineering*, 34(5):415–430, 2019. 2
- [28] Chul Min Yeum, Jongseong Choi, and Shirley J Dyke. Automated region-of-interest localization and classification for vision-based visual assessment of civil infrastructure. *Structural Health Monitoring*, 18(3):675–689, 2019. 2
- [29] Yasutaka Narazaki, Vedhus Hoskere, Tu A Hoang, Yozo Fujino, Akito Sakurai, and Billie F Spencer Jr. Vision-based automated bridge component recognition with high-level scene consistency. *Computer-Aided Civil and Infrastructure Engineering*, 35(5):465–482, 2020. 2
- [30] Andrey Dimitrov and Mani Golparvar-Fard. Vision-based material recognition for automated monitoring of construction progress and generating building information modeling from unordered site image collections. *Advanced Engineering Informatics*, 28(1):37–49, 2014. 2
- [31] Mani Golparvar-Fard, Feniosky Pena-Mora, and Silvio Savarese. Automated progress monitoring using unordered daily construction photographs and ifc-based building information models. *Journal of Computing in Civil Engineering*, 29(1):04014025, 2015. 2
- [32] Hesam Hamledari, Shakiba Davari, Ehsan Rezazadeh Azar, Brenda McCabe, Forest Flager, and Martin Fischer. Uav-enabled site-to-bim automation: Aerial robotic-and computer vision-based development of as-built/as-is bims and quality control. In *Construction research congress*, pages 336–346, 2017. 2
- [33] Patrick Aravena Pelizari, Christian Geiß, Paula Aguirre, Hernán Santa María, Yvonne Merino Peña, and Hannes Taubenböck. Automated building characterization for seismic risk assessment using street-level imagery and deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 180:370–386, 2021. ISSN 0924-2716. <https://doi.org/https://doi.org/10.1016/j.isprsjprs.2021.07.004>. URL <https://www.sciencedirect.com/science/article/pii/S0924271621001817>. 2
- [34] Daniela Gonzalez, Diego Rueda-Plata, Ana B. Acevedo, Juan C. Duque, Raúl Ramos-Pollán, Alejandro Betancourt, and Sebastian García. Automatic detection of building typology using deep learning methods on street level images. *Building and Environment*, 177:106805, 2020. ISSN 0360-

1323. <https://doi.org/https://doi.org/10.1016/j.buildenv.2020.106805>.
URL <https://www.sciencedirect.com/science/article/pii/S0360132320301633>. 2
- [35] Jürgen Hackl, Bryan Adey, Michał Woźniak, and Oliver Schümperlin. Use of unmanned aerial vehicle photogrammetry to obtain topographical information to improve bridge risk assessment. *Journal of Infrastructure Systems*, 24:04017041, 03 2018. [https://doi.org/10.1061/\(ASCE\)IS.1943-555X.0000393](https://doi.org/10.1061/(ASCE)IS.1943-555X.0000393). 2
- [36] Mahmoud Miari, Kok Keong Choong, and Robert Jankowski. Seismic pounding between adjacent buildings: Identification of parameters, soil interaction issues and mitigation measures. *Soil Dynamics and Earthquake Engineering*, 121:135–150, 2019. ISSN 0267-7261. <https://doi.org/https://doi.org/10.1016/j.soildyn.2019.02.024>.
URL <https://www.sciencedirect.com/science/article/pii/S0267726118313848>. 3
- [37] Jorge Aguilar Carboney, Hugón Juárez García, Rodolfo Ortega, and Jesús Iglesias. The mexico earthquake of september 19, 1985 - statistics of damage and of retrofitting techniques in reinforced concrete buildings affected by the 1985 earthquake. *Earthquake Spectra*, 5, 02 1989. <https://doi.org/10.1193/1.1585516>. 3
- [38] Kazuhiko Kasai and Bruce F. Maison. Building pounding damage during the 1989 loma prieta earthquake. *Engineering Structures*, 19(3):195–207, 1997. ISSN 0141-0296. [https://doi.org/https://doi.org/10.1016/S0141-0296\(96\)00082-X](https://doi.org/https://doi.org/10.1016/S0141-0296(96)00082-X). URL <https://www.sciencedirect.com/science/article/pii/S014102969600082X>. 3
- [39] Jeng-Hsiang Lin and Cheng-Chiang Weng. A study on seismic pounding probability of buildings in taipei metropolitan area. *Journal of the Chinese Institute of Engineers*, 25(2):123–135, 2002. <https://doi.org/10.1080/02533839.2002.9670687>. URL <https://doi.org/10.1080/02533839.2002.9670687>. 3
- [40] Hemant B. Kaushik, Kaustubh Da, Dipti Ranjan Sahoo, and Gayatri Kharel. Performance of structures during the sikkim earthquake of 14 february 2006. *Current Science*, 91:449–455, 2006. 3
- [41] Nurullah Bektaş and Orsolya Keyes-Brassai. Conventional rvs methods for seismic risk assessment for estimating the current situation of existing buildings: A state-of-the-art review. *Sustainability*, 14(5), 2022. ISSN 2071-1050. <https://doi.org/10.3390/su14052583>. URL <https://www.mdpi.com/2071-1050/14/5/2583>. 3
- [42] Pradeep Ramancharla, Aniket Bhalkikar, Pulkit Velani, Pammi Vyas, Bharat Prakke, Neelima Patnala, and Niharika Talyan. A primer on rapid visual screening (rvs) consolidating earthquake safety assessment efforts in india. 10 2020. 3
- [43] Giuseppina Vacca, Andrea Dessì, and Alessandro Sacco. The use of nadir and oblique uav images for building knowledge. *ISPRS International Journal of Geo-Information*, 6:393, 12 2017. <https://doi.org/10.3390/ijgi6120393>. 3

- [44] Christopher Arnold and Robert Reitherman. *Building configuration and seismic design*. John Wiley & Sons, 1982. 3
- [45] Liora Sahar, Subrahmanyam Muthukumar, and Steven P French. Using aerial imagery and gis in automated building footprint extraction and shape recognition for earthquake risk assessment of urban inventories. *IEEE Transactions on Geoscience and Remote Sensing*, 48(9):3511–3520, 2010. 3
- [46] Dong Chen, Liqiang Zhang, Jonathan Li, and Rei Liu. Urban building roof segmentation from airborne lidar point clouds. *International Journal of Remote Sensing*, 33(20):6497–6515, 2012. <https://doi.org/10.1080/01431161.2012.690083>. URL <https://doi.org/10.1080/01431161.2012.690083>. 3, 4
- [47] Anh-Vu Vo, Linh Truong-Hong, Debra F. Laefer, and Michela Bertolotto. Octree-based region growing for point cloud segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 104:88–100, 2015. ISSN 0924-2716. <https://doi.org/https://doi.org/10.1016/j.isprsjprs.2015.01.011>. URL <https://www.sciencedirect.com/science/article/pii/S0924271615000283>. 4
- [48] Aparajithan Sampath and Jie Shan. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 48(3):1554–1567, 2010. <https://doi.org/10.1109/TGRS.2009.2030180>. 4
- [49] Zhen Dong, Bisheng Yang, Pingbo Hu, and Sebastian Scherer. An efficient global energy optimization approach for robust 3d plane segmentation of point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 137:112–133, 2018. ISSN 0924-2716. <https://doi.org/https://doi.org/10.1016/j.isprsjprs.2018.01.013>. URL <https://www.sciencedirect.com/science/article/pii/S0924271618300133>. 4
- [50] Weijia Li, Conghui He, Jiarui Fang, Juepeng Zheng, Haohuan Fu, and Le Yu. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source gis data. *Remote Sensing*, 11(4), 2019. ISSN 2072-4292. <https://doi.org/10.3390/rs11040403>. URL <https://www.mdpi.com/2072-4292/11/4/403>. 4
- [51] Yuchu Qin, Yunchao Wu, Bin Li, Shuai Gao, Miao Liu, and Yulin Zhan. Semantic segmentation of building roof in dense urban environment with deep convolutional neural network: A case study using gf2 vhr imagery in china. *Sensors*, 19:1164, 03 2019. <https://doi.org/10.3390/s19051164>. 4
- [52] Aathreya S. Bhat, Amith V. Shivaprakash, Namrata S. Prasad, and Chaitra Nagaraj. Template matching technique for panoramic image stitching. In *2013 7th Asia Modelling Symposium*, pages 111–115, 2013. <https://doi.org/10.1109/AMS.2013.22>. 4
- [53] Somaya Adwan, Iqbal Alsaleh, and Rasha Majed. A new approach for image stitching technique using Dynamic Time Warping (DTW) algorithm towards scoliosis X-ray diagnosis. *Measurements*, 84:32–46, April 2016. <https://doi.org/10.1016/j.measurement.2016.01.039>. 4

- [54] Moushumi Bonny and Mohammad Uddin. A technique for panorama-creation using multiple images. *International Journal of Advanced Computer Science and Applications*, 11, 01 2020. <https://doi.org/10.14569/IJACSA.2020.0110293>. 4
- [55] Murtadha Alomran and Douglas Chai. Feature-based panoramic image stitching. In *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1–6, 2016. <https://doi.org/10.1109/ICARCV.2016.7838721>. 4
- [56] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov 2004. ISSN 1573-1405. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>. URL <https://doi.org/10.1023/B:VISI.0000029664.99615.94>. 4
- [57] Ying Zhang, Lei Yang, and Zhujun Wang. Research on video image stitching technology based on surf. In *2012 Fifth International Symposium on Computational Intelligence and Design*, volume 2, pages 335–338, 2012. <https://doi.org/10.1109/ISCID.2012.235>. 4
- [58] Paul Tsao, Tsi-Ui Ik, Guan-Wen Chen, and Wen-Chih Peng. Stitching aerial images for vehicle positioning and tracking. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 616–623. IEEE, 2018. 4
- [59] Yilin Liu, Fuyou Xue, and Hui Huang. Urbanscene3d: A large scale urban scene dataset and simulator. 2021. 6
- [60] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019. 6
- [61] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 7
- [62] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 7
- [63] Yu Wang, Quan Zhou, Jia Liu, Jian Xiong, Guangwei Gao, Xiaofu Wu, and Longin Jan Latecki. Lednet: A lightweight encoder-decoder network for real-time semantic segmentation. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1860–1864, 2019. <https://doi.org/10.1109/ICIP.2019.8803154>. 8
- [64] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *arXiv preprint arXiv:2201.02605*, 2021. 10