Ground Penetration Depth Measurement of Driven Piles Using Optical Flows and Deep Pose Estimation

Kornvik Tanpipat
Stanford University
kornvik@stanford.edu

Abstract

Measuring the ground penetration depth of driven piles is a critical task in civil engineering. Traditional methods are labor-intensive and imprecise, while other alternatives require the setup or use of expensive tools. This study explores two approaches leveraging classical computer vision and deep learning techniques: optical flow and one-shot deep pose estimation for feature tracking and camera pose estimation. Our approach focuses on a practical method in Thailand, measuring pile ground penetration depth after every ten hammer blows. We record video data and ground truth depth measurements from a construction site in Thailand to validate our techniques.

Our results show that optical flow achieves tracking accuracy within a 32.8% margin of error in real-world conditions. Additionally, under simulation, our deep learning model demonstrates a tracking accuracy with a margin of error of 86.1%, though this increases to 286% in the field. These findings suggest that while optical flow is more reliable under practical conditions, our deep-learning approach shows potential in controlled environments but requires further development for real-world application. This project lays the groundwork for developing a low-cost, practical solution that can be readily adopted in the field, with future work aimed at improving the real-world performance of the deep learning model.

1. Introduction

In civil engineering and construction, piles serve as structural pillars typically made of concrete or steel, driven deep into the ground to provide essential support and ensure the stability of deep foundation infrastructure. A widely adopted method for driving a pile into the ground involves repeatedly dropping a large hammer or ram onto the top of the pile, as shown in Figure 1.

To assess the stability of the driven pile, various indicators must be verified. One key criterion is the blow count, which quantifies the effort required to advance a pile into the ground and is measured as the number of hammer impacts needed for the pile to penetrate a specific distance. However, conventional methods for measuring pile penetration depth often involve manual procedures that can be time-consuming, labour-intensive, and imprecise. Alternative methods, such as accelerometers, are considered expensive, prone to break due to the impulse force of the hammer and may require additional setup aids.

Figure 1. Image of a pile being driven into the ground by a hammer.

To address these challenges, this project investigates two different methods: classical computer vision techniques, primarily optical flow, and deep pose estimation for feature tracking and camera pose estimation, to develop efficient, accurate, and low-cost solutions for pile depth measurement. Additionally, we compare the results between the two approaches and identify the most effective approach for practical deployment in construction sites.

2. Related Work

Several studies have explored innovative methods for pile position and pose estimation. Youwai-Makam [6] employed computer vision and artificial markers for dynamic pile load tests, emphasizing pile capacity estimation but introducing complexity with markers. Meanwhile, Tong *et*

al. [3] developed a laser range finder-based navigation system for precise pile driving.

In contrast, our project enhances pile depth measurement accuracy and efficiency using 3D image analysis and computer vision based on video data, targeting blow count measurements. Our approach, while sharing similarities with Youwai-Makam, aims to eliminate the need for markers and focuses on blow count measurement.

3. Technical Approach

The project revolves around finding a camera pose to estimate pile movement based on calibrated images and identified feature points. Using the pile's known industry-standard dimensions, we track its movement during the video scene.

3.1. Feature Points Tracking

3.1.1 Camera Pose Estimation Using Optical Flow

Our initial method involves feature tracking using the optical flow technique, starting with pre-identified feature points in the first frame of each scene. The Lucas-Kanade optical flow algorithm is utilized to track the movement of points between consecutive frames. Based on the feature points and the known dimensions of the pile, as illustrated in Figure 2, we can estimate the camera pose, R and t. We solve this using the AP3P algorithm by Ke-Roumeliotis [1], an efficient nonlinear approach to the perspective-three-points problem.

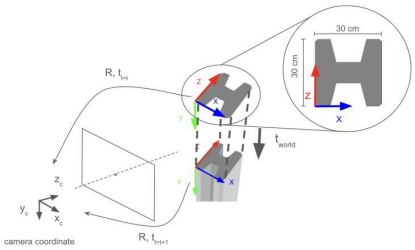

Figure 2. Camera and pile system coordinate setup with translation vector.

3.1.2 Feature Tracking Using Deep Object Pose Estimation Model

As an alternative, we investigate training a deep convolutional network to track feature points between frames. We base our approach on the Deep Object Pose Estimation Model by Jonathan et al. [5].

We train the network to identify feature points rather than directly estimating the camera orientation to avoid issues related to discontinuities in orientation representation [4]. By focusing on feature point identification, we can subsequently apply the solvePnP algorithm to determine the camera orientation, ensuring a more robust and accurate estimation process. This approach mitigates the complexities and potential errors associated with directly training a model to estimate camera orientation.

Data Generation

We create a detailed 3D model of the pile in a simulation environment using Blender. This model replicates the pile's actual dimensions and appearance used on construction sites. By rendering images of the pile from various angles and positions and augmenting images with different backgrounds and occlusions, we generate a comprehensive dataset capturing different perspectives with varying distances between 1 and 2 meters and different lighting conditions. To avoid ambiguity due to the symmetry of the pile object, we only generated data from one side of the pile by constraining the camera to the positive y-axis, as shown in Figure 3.

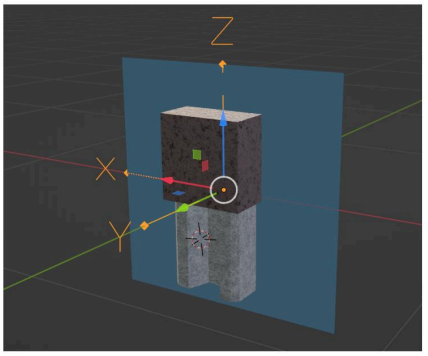

Figure 3. Pile ambiguity along y plane

Model Architecture

We use these rendered images to train the deep convolutional neural network. The training process involves feeding the model images of the rendered pile along with the corresponding belief maps, which are generated from feature points. This enables the network to learn the relationship between the pile's visual appearance and the corresponding feature points' positions.

The overall architecture is illustrated in Figure 4. The input image is first scaled to 400×400 and inputted into the network. The network processes data in multiple stages, where each stage considers both the image features and the

outputs from the previous stage and outputs the belief maps, each representing the feature points' location. As each stage is convolutional, the effective receptive field grows larger with each stage, allowing the network to incorporate more context and resolve ambiguities present in the earlier stages with smaller receptive fields.

First, we leverage transfer learning by using the pre-trained VGG-19 model to extract features from its first 24 layers, followed by two 3×3 convolution layers. These 128-dimensional features are then fed into each of the subsequent stages of the model.

The first stage of the network consists of three 3×3×128 layers and one 1×1×512 layer, followed by a 1×1×4 layer to produce belief maps.

The remaining five stages are similar to the first stage but leverage larger filter sizes, allowing the model to understand higher-level features from the image input. Each stage receives the belief map from the previous stage and the extracted features, resulting in an input dimension of 132 (128 + 4).

Each of these stages consists of five 7×7×128 layers and one 1×1×128 layer, ending with a 1×1×4 layer. ReLU activation functions are used throughout the network. The belief map outputs for each stage from the 1st to the 6th are of size 4 x 50 × 50.

After the network processes an image, we calculate the coordinates of each vertex by searching for local peaks in each belief map, computing the weighted average to determine the coordinates, and scaling the coordinates up to the original image size.

Model Training

3.2. Pile Movement Estimation

Using the two different methods above, we are able to estimate the rotation matrix and translation vector. Under the assumption that the pile does not rotate, we can estimate the pile movement from the difference in the re-projected translation vector on the pile coordinate system, as shown in Figure 2. Since the y-axis represents the depth of the pile toward the ground, we can retrieve the y-axis from the second element of the $\Delta \mathbf{t}_{world}$.

$$\Delta \mathbf{t}_{world} = \mathbf{R}^{-1} \Delta \mathbf{t} = \begin{bmatrix} \Delta t_{x\_world} \\ \Delta depth \\ \Delta t_{z\_world} \end{bmatrix}$$

3.3. Improving accuracy

3.3.1 Pile Segmentation

Optical flow is highly susceptible to noise, which can significantly affect the accuracy of feature tracking and camera pose estimation. To mitigate this issue and simplify the

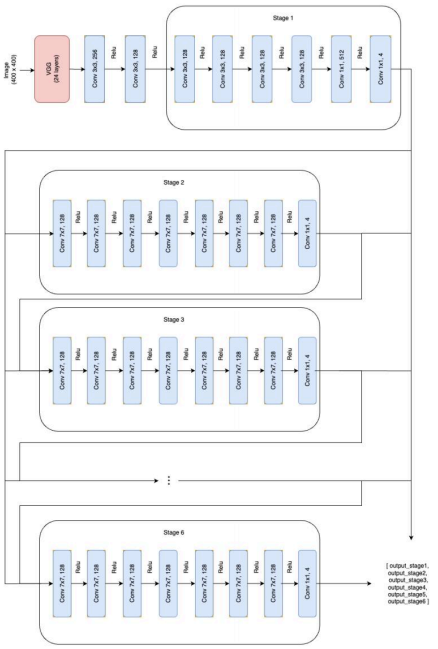

Figure 4. Deep Object Pose Estimation model architecture

complexity of this project, we preprocess the images by segmenting out the pile using a pre-trained Segment Anything Model (SAM) by Kirillov et al. [2]. This segmentation step helps isolate the pile from the background and other objects in the scene, reducing noise and enhancing the accuracy of the optical flow algorithm.

We are able to utilize the SAM model out of the box because we assume the pile is consistently located in the center of the image as shown in Figure 5. This assumption simplifies the segmentation process, allowing SAM to effectively and reliably isolate the pile without requiring additional modifications or training.

The segmented images are also used as input for the pose estimation model. By providing a cleaner and more focused input, the segmentation simplifies the training process and allows the model to learn better the relationships between the pile's visual appearance and the corresponding feature points' positions. This preprocessing step ensures that both the optical flow and pose estimation models work with high-quality, relevant data, ultimately improving the overall accuracy of our approach.

4. Experiments

4.1. Data Collection

Videos were collected at a construction site in Thailand using the Intel RealSense D435i camera. Depth data was

Figure 5. Segmentation result example based on pile position assumption

also collected and is not used as we aim to estimate the pile movement solely on RGB images. We have collected 12 scenes from the construction site for this project, along with corresponding pile depth estimation for each scene. Additionally, the initial setup involved manually identifying the pile's corner pixels in each video's first frame.

4.2. Data Generation

To train the deep pose network, we generated a synthetic dataset using a detailed 3D model of the pile created in Blender. This dataset includes 12,000 rendered images of the pile from various angles and positions, augmented with different translations, rotations, scales, and occlusions to simulate real-world conditions, as shown in Figure 6.

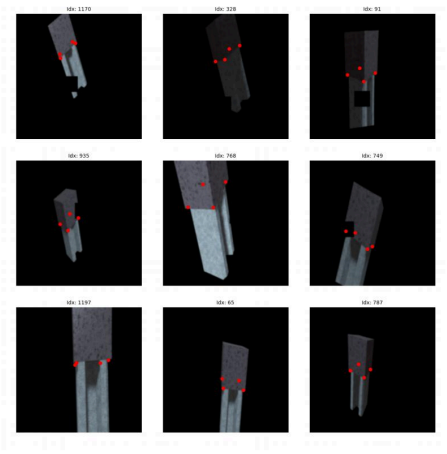

Figure 6. Generated with data augmentation

4.3. Feature Tracking with Optical Flow

We tested feature tracking algorithms using optical flow. This method works well using pre-identified feature points, as shown in Figure 7.

Figure 7. Sample frame from collected video data with tracked features (red dots with white trails).

However, without the additional segmentation, the method also has its limitations where tracking could easily fail to keep tracing the feature points, resulting in unusable data for pile movement estimation, as shown in Figure 8.

Figure 8. Failure frame from collected video data with tracked features in Scene 4 (red dots with white trails).

By applying segmentation, we were able to mitigate the issue substantially, as shown in Figure 9. Segmentation helps isolate the pile from the background, reducing the impact of occlusion and varying illumination on feature tracking.

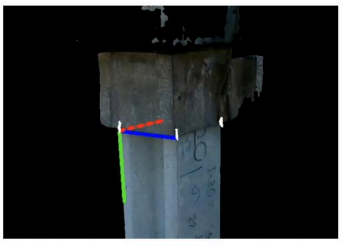

Figure 9. Frame from Scene4 result with segmentation applied (red dots with white trails.

4.3.1 Optical Flow Measurement Error

The measurement errors are calculated by comparing estimated depths to manual measurements averaged among the 12 recorded scenes using formula:

$$\text{Error } \% = \left(\frac{|\text{Estimated Depth} - \text{Manual Depth}|}{\text{Manual Depth}} \right) \times 100$$

By using the above measurement error equation, we are able to compare the result of optical flow approaches in Table 1.

Method	Measurement Error
Optical Flow without Segmentation	118%
Optical Flow with Segmentation	32.8%

Table 1. Pile Depth Measurement Error With and Without Segmentation

4.4. Feature Tracking with Deep Object Pose Estimation (DOPE)

DOPE offers greater flexibility since it eliminates the need for manually pre-identifying feature points on a pile.

4.4.1 Model Training

The deep convolutional neural network's training process involves feeding the model images of the rendered pile along with the corresponding belief maps generated from feature points. The model was trained using a supervised learning approach, with the belief maps serving as the ground truth.

We utilized the Adam optimizer with a learning rate of 0.001 and a batch size of 16. Training was conducted over 25 epochs, with early stopping based on the validation loss to prevent overfitting. A learning rate scheduler, specifically `optim.lr_scheduler.ReduceLROnPlateau`, was applied with parameters `patience=5` and `factor=0.1` to reduce the learning rate when the validation loss plateaued. Data augmentation techniques, including translation, rotation, and scaling, were applied during training to enhance the model's robustness to various transformations.

The loss function used was mean squared error (MSE) for the belief maps. The total loss was aggregated by calculating the loss against each of the intermediate stage's belief maps. Figure 10 shows the training and validation loss plot for training the network, which was split into training, validation, and testing sets with a ratio of 70-15-15, ensuring a diverse range of perspectives and lighting conditions in each set.

We observed that the model could identify feature points from generated images in the test group, as shown in Figure 11, where the dimmed dots represent the ground truth positions of the feature points and the bright dots represent the estimated points.

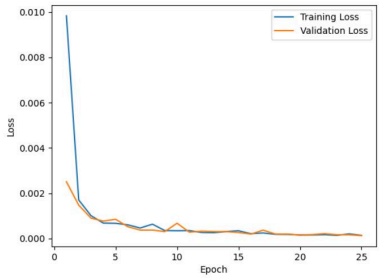

Figure 10. Training and validation loss against epoch

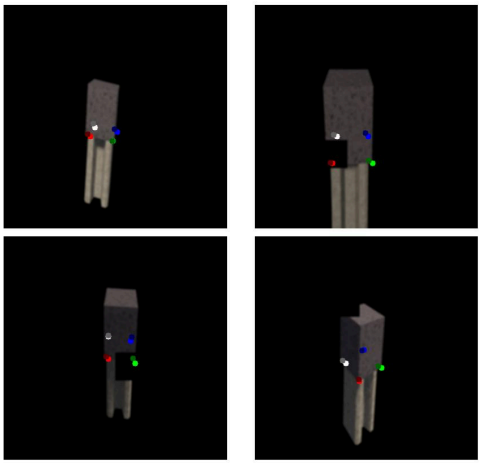

Figure 11. Feature points identified by the trained model on test images (bright points) against ground truth feature points (dimmed points).

However, when testing against real scenes, we found that while DOPE provides a robust initial framework for feature tracking, it does not output stable results across video frames. This instability resulted in a measurement error of 286% calculated in 7 out of 12 scenes, as can be seen in Figure 12. Additionally, the model failed to recognize the pile in the remaining 5 scenes.

As a result, we generated animated sequences for three scenes with varying perspectives and ground truth depth penetration instead of using real scenes to compare the results with the optical flow approach consistently. Figures 13 show the sampled frame from these three generated scenes.

From the three generated scenes, we report the measurement error results in Table 2.

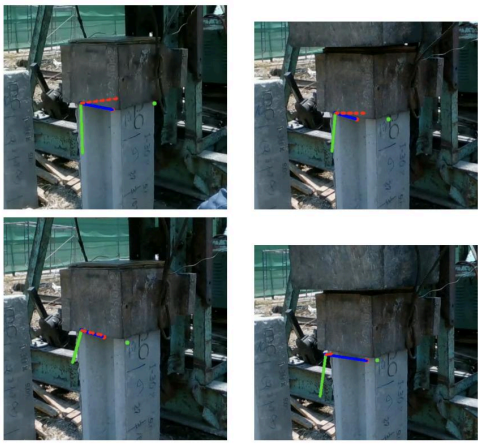

Figure 12. Example results in Scene 4 using the trained model to track feature points

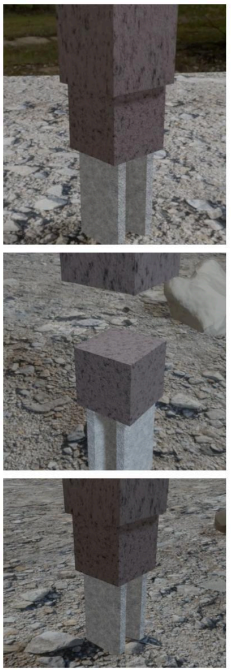

Figure 13. Sampled frame from each simulated scene

5. Discussion

The experiments conducted in this study highlight several critical findings regarding the estimation of pile movement using RGB images. The use of optical flow with and without segmentation presents a significant contrast in performance. Without segmentation, the feature tracking often fails due to occlusion and varying illumination, resulting in high measurement errors. Applying segmentation improves the robustness of feature tracking by isolating the pile from the background, significantly reducing the error rate.

Method	Measurement Error
Optical Flow with Segmentation	40.4%
DOPE with Segmentation	86.1%

Table 2. Pile Depth Measurement Error Comparing Optical Flow and DOPE

The deep object pose estimation (DOPE) approach offers a promising alternative by eliminating the need for manual feature point identification. While DOPE shows robust performance on synthetic data, its application to real-world scenes presents challenges. The instability of the DOPE model across video frames and its failure to recognize the pile in certain scenes highlight the need for further improvements in real-world adaptability.

Additionally, the substantial difference in measurement errors between optical flow with segmentation (32.8%) and without segmentation (118%) underscores the importance of integrating segmentation into feature tracking methodologies. The higher error rate of DOPE (286%) in real scenes indicates that while deep learning models hold potential, their deployment in real-world environments requires additional tuning and validation.

6. Conclusion and Future Work

In this study, we explored the use of optical flow and deep object pose estimation (DOPE) for estimating pile movement from RGB video data. Our findings demonstrate that segmentation significantly enhances the performance of optical flow by improving feature tracking accuracy. While DOPE shows promise, it currently faces challenges in generalizability toward real-world applications, with high error rates and instability across frames.

For future work, we propose the following directions to address these challenges:

1. **Integration of Kalman Filters:** Implementing a Kalman filter to smooth and predict the movement of the pile based on the tracked feature points could significantly enhance the stability and accuracy of both optical flow and DOPE approaches. Kalman filters are well-suited for handling the noise and uncertainties present in real-world data, providing a robust framework for continuous tracking. We observed many jumps in the video predictions made by DOPE, and believe that incorporating a Kalman filter will substantially improve the accuracy and consistency of the results.

2. **Improvement of DOPE Model:** Further training of the DOPE model with more diverse real-world data and augmentations could improve its performance. Incorporating more varied lighting conditions, backgrounds, and occlusions in the training dataset can enhance the model's robustness.

3. **Hybrid Approach:** Combining the strengths of optical flow with segmentation and DOPE could yield a more reliable solution. For instance, using DOPE for initial feature point detection followed by optical flow tracking with segmentation might leverage the benefits of both methods.

By addressing these areas, we aim to develop a more robust and accurate system for estimating pile movement from RGB video data, ultimately contributing to improved monitoring and safety in construction projects.

Supplementary Material

Additional resources, including code and data used in this study, are available on our GitHub repository. The repository contains detailed instructions for reproducing the experiments and further exploring the methodologies discussed in this project. In addition, we provide GIF images demonstrating how pile tracking works with different approaches on the README. You can access it at the following link:
https://github.com/kornvik/cs231_project

Acknowledgements

We thank the construction site team in Thailand for their cooperation and support in data collection. We also express our gratitude to the teaching assistants and instructors of the CS231A class of 2024 for their invaluable guidance, feedback, and support throughout this project. Their insights and expertise have been instrumental in shaping our research and enhancing our understanding of the subject matter.

References

[1] Tong Ke and Stergios Roumeliotis. An efficient algebraic solution to the perspective-three-point problem. 01 2017. 2

[2] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 3

[3] Takeshi Sasaki, Hiroshi Kawahara, Fumihiro Inoue, and Hideki Hashimoto. Circle fitting based pile positioning and machine pose estimation from range data for pile driver navigation. *IFAC Proceedings Volumes*, 45(22):848–853, 2012. 10th IFAC Symposium on Robot Control. 2

[4] Ashutosh Saxena, Justin Driemeyer, and Andrew Y. Ng. Learning 3-d object orientation from images. In *2009 IEEE International Conference on Robotics and Automation*, pages 794–800, 2009. 2

[5] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox, and Stan Birchfield. Deep object pose estimation for semantic robotic grasping of household objects, 2018. 2

[6] Sompote Youwai and Parchya Makam. Ctrpile: A computer vision and transformer approach for pile capacity estimation from dynamic pile load test. In *The 2024 IEEE Conference on Artificial Intelligence (IEEE CAI 2024)*, Singapore, March 2024. 1