Synthetic Data Generation and Visual Servoing Simulation for Vision-Guided Robotic Rebar Tying

Chethiya Prasanga Wadumesthri[1] and Aritra Pal[1]

[1] Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, India
ce25D802@smail.iitm.ac.in , aritrapal@civil.iitm.ac.in

Abstract

Automatic rebar tying using a vision-guided robotic system requires a reliable perception of rebar intersections within densely arranged rebar meshes. However, major challenges include the scarcity of large-scale, precisely annotated visual datasets for training perception models and the limited evaluation of models trained on synthetic data in the context of robot alignment through visual servoing. This paper proposes an end-to-end framework for synthetic data generation and perception model evaluation that integrates parametric modeling, physics-based rendering, deep learning–based rebar intersection detection, and visual servoing simulation for robotic alignment. Perception models trained exclusively on the synthetic data are deployed in a robotic simulation environment and evaluated using both image-based visual servoing (IBVS) and position-based visual servoing (PBVS) with a camera-mounted setup. The results demonstrate that the trained models effectively to the servoing task, enabling the accurate identification of rebar intersection points and the precise alignment of the tool center point (TCP) without requiring manual data collection or labeling. This study validates synthetic data as a viable alternative to real-world annotation for vision-guided robotic systems in complex construction scenarios.

Keywords –

Rebar tying; Synthetic data; Rebar intersection; Visual servoing; Vision-guided robotics

1 Introduction

Studies project that by 2050, 68% of the global population will live in urban areas, intensifying infrastructure demands in cities worldwide [1]. As metropolitan suburbs expand to accommodate this growth, the demand for large-scale urban infrastructure has increased significantly [2]. Reinforced cement concrete (RCC) has therefore become a fundamental construction material, accounting for a substantial share of infrastructure development costs and serving as a primary load-bearing system in buildings and civil structures [3]. RCC elements must safely withstand tensile forces and effectively redistribute structural loads, a requirement that is achieved using steel reinforcement bars (rebar). Rebar provides the necessary tensile capacity and ensures structural integrity and durability under service and extreme loading conditions [4]. Consequently, reliable rebar mesh connections—achieved through tying, welding, or mechanical fastening are essential to reinforced concrete construction [5], with rebar tying being the most widely used method for securing intersecting bars with wire [6]. However, manual rebar tying is one of the most physically demanding activities in concrete construction, involving high labor requirements and significant ergonomic risks [7]. Such manual-dependent operations impose substantial strain on workers, contribute to inefficient work practices, and can delay overall project progress [8]. Additionally, variability in manual tying can lead to quality control issues that directly affect structural performance. To overcome these challenges, rebar tying robots have been developed to automate the tying of pre-positioned reinforcement, reducing reliance on manual labor while improving productivity and consistency[9].

To automate rebar tying tasks, various robotic systems have been developed. Commercial solutions include gantry-mounted robots such as IronBot and TyBot [6]. However, these systems are primarily suited to linear construction projects (e.g., bridge decks and highways), are costly and time-intensive to install, and require specialized operators [8]. To address these limitations, mobile rebar tying robots such as those developed by the Beijing Institute of Technology and Taisei Corporation’s T-iROBO integrate a mobile base, robotic arm, and adjustable end effectors for tying operations[10]. To enhance operational efficiency, many of these systems incorporate vision-based perception, typically using a 3D camera mounted near the end effector to capture visual data and estimate the robot arm pose through visual perception algorithms [11]. However, vision-based rebar detection is significantly affected by harsh construction site conditions. Complex lighting, steel reflections, and cluttered backgrounds degrade algorithm performance and complicate reliable rebar recognition [12] [13]. Background interference and lighting variability can lead to false positives, inconsistent segmentation of rebar edges, missed detections, and incomplete identification of rebar intersections [13]. To overcome these challenges, visual servoing has emerged as a critical control strategy for rebar tying robots. Visual servoing systems use camera feedback to guide the robot toward a desired configuration [14]. Two main approaches have been proposed: position-based visual servoing (PBVS) and image-based visual servoing (IBVS) [15]. PBVS defines the control strategy reference in 3D Cartesian space by estimating the camera–target pose from image data, whereas IBVS operates directly in the image plane, minimizing the error between current and desired image features represented as low-dimensional visual primitives.

Accurate target object detection is essential for visual servo–based robotic systems. State-of-the-art deep learning approaches, particularly convolutional neural networks (CNNs), have demonstrated superior performance in object recognition [16], camera relocalization [17], and object pose estimation [18]. More recently, CNNs have been integrated into visual servoing frameworks to address limitations of traditional methods [19]. Liu et al. proposed a vision-based six-degree-of-freedom rebar tying robot with a novel tying algorithm; however, the system does not employ a closed-loop visual servoing framework for motion control and is limited to vertical rebar meshes [11]. The effectiveness of deep learning models strongly depends on the quantity and quality of labeled training data [20] . To increase data, researchers use a BIM-based pipeline. However, lighting and texture variations remain challenging. Consequently, researchers often rely on publicly available datasets such as COCO [21], ACID [22], and PEER Hub ImageNet [23]. When suitable datasets are unavailable, manual data annotation becomes necessary. This poses a significant challenge for rebar mesh applications, as creating high-quality datasets—especially for dense meshes with numerous rebar intersections—is costly and time-consuming, thereby limiting the practical deployment of deep learning–based approaches.

In this paper, we make two contributions firstly a domain-specific synthetic data generation pipeline for curved rebar mesh geometries, including three custom Blender plugins: the Lighting Kit Plugin (LKP), Material Kit Plugin (MKP), and Rebar Annotator Plugin, which allows keypoint annotation to be completely automated by 3D geometric reasoning and does not require manual labelling. Secondly, we present an end-to-end pipeline for generating synthetic datasets and visually evaluating perception models trained on this synthetic data for detecting intersections in rebar mesh. This pipeline is integrated into a closed-loop visual servoing framework, allowing robots to move toward intersection points in the rebar mesh. Our proposed approach highlights the convergence of IBVS and the effectiveness of PBVS strategies for accurate targeting and control.

2 Method

Rebar intersection identification using CNN-based key-point detection models typically require large amounts of training data to effectively learn the geometric and visual properties of intersection objects. To address this requirement, we develop an end-to-end synthetic visual data generation and perception model evaluation pipeline. As illustrated in Figure 1, the proposed pipeline uses Revit and Blender to generate realistic synthetic datasets while automatically producing ground-truth annotations. The dataset accurately represent rebar mesh configurations in reinforced concrete structures. The pipeline consists of four main stages: (1) parametrized 3D scene modelling, (2) physics-based rendering, (3) rebar intersection detection model training and evaluation, and (4) IBVS and PBVS simulation using the Webots robot simulator [24] .

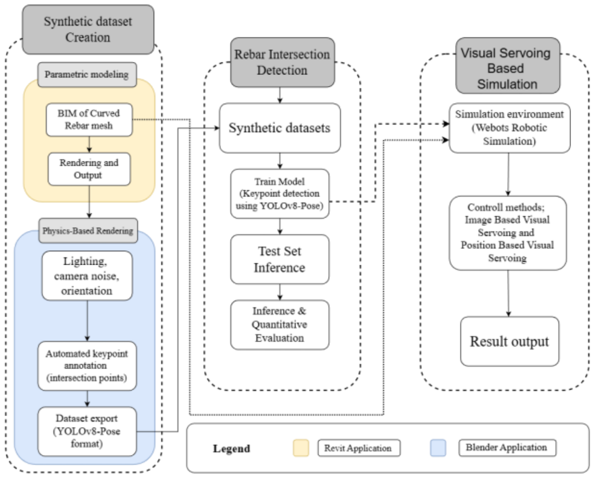

Figure 1. End-to-end framework for synthetic dataset generation, rebar mesh intersection detection, and visual servoing simulation

2.1 Synthetic dataset creation

Synthetics dataset generation pipeline consists of two steps: (a) parametric modelling and physics-based rendering and (b) automatic key-point annotation and dataset creation.

2.1.1 Parametric modelling and physic-based rendering

As shown in Figure 2, a parametric modelling approach is adopted in Autodesk Revit, where rebar mesh geometry is defined by a diameter of 12mm, bar spacing (100 mm), and tunnel segment dimensions intrados arc length 2281.24 mm, extrados arc length 2513.01 mm, and width 3019.43 mm. The resulting rebar mesh, particularly the top reinforcement layer, is subsequently imported into Blender to serve as a foundation for further parametric modelling and processing, as illustrated in Figure 3. To generate a diverse range of rebar appearances and background environments, we systematically randomize visual parameters, including base textures, background surfaces, lighting conditions, and camera angles, utilizing custom-made Blender plugins: the LKP and the MKP. This approach ensures a comprehensive, detailed exploration of the visual possibilities in our project.

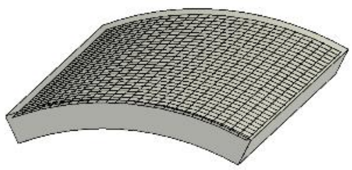

Figure 2. Created 3D Rebar tunnel Segment

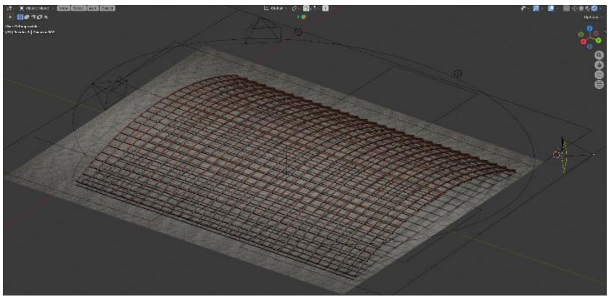

Figure 3. Tunnel rebar mesh placements

The LKP is a tool based on Blender, designed to create lighting configurations for the generation of synthetic rebar mesh data. As illustrated in Figure 4a, it offers a cohesive interface for managing various illumination parameters, including light intensity, color temperature, light types (such as sun, area, and point), and the direction of sunlight through azimuth and elevation controls. The ability to adjust shadow softness allows for realistic variations in the environment, facilitating the creation of consistent yet diverse lighting conditions for large-scale synthetic dataset generation.

To introduce realistic material variability in the synthetic rebar dataset, a custom MKP was developed in Blender, aligning with the design principles of the proposed LKP. The MKP facilitates systematic randomization of the rebar surface appearance by enabling users to upload and apply a diverse range of texture maps to rebar meshes. These textures reflect realistic material conditions, such as corrosion, surface wear, and manufacturing variations, and are sourced from the open-source asset library, Poly Haven. The plugin offers an intuitive interface for efficient texture selection and application, significantly enhancing the visual diversity of the generated synthetic data. Figure 4b showcases the MKP interface along with examples of the material variations applied to rebar meshes.

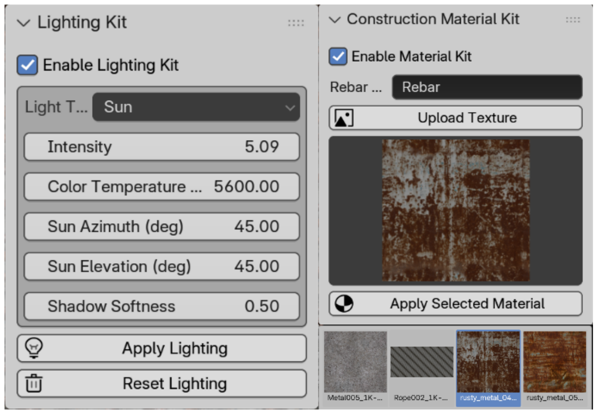

Figure 4. (a) Configurable lighting module for synthetic rebar dataset illumination control. (b) Material configuration panel for assigning rebar surface textures

2.1.2 Automatic annotation and synthetic dataset creation.

Further to the parametric modelling and rendering, a Rebar Annotator blender plugin was developed to facilitate rule-based rebar intersection annotation. The plugin first identifies rebar intersection points using geometric reasoning in three-dimensional space and subsequently generates key-point annotations using 3D to 2D projections. Annotations were further exported into YOLO-key point detection format.

Figures 5 illustrates custom Blender plugin developed to generate large-scale annotated synthetic datasets. This plugin offers configurable rendering parameters, including options for image resolution, frame sampling rate, and output limits, which enable controlled generation of diverse training data. For each rendered image, annotations that are ready for machine learning and compatible with Ultralytics YOLO (UY) are automatically generated. The pipeline supports two types of annotation formats: object detection and key-point estimation. A key-point annotation example is shown in Figure 6.

The synthetic dataset was generated using the configuration depicted in Figure 7. This configuration includes a rebar mesh, three light sources, and three distinct camera trajectories. Photorealistic image rendering was executed using Blender's Cycles renderer, which utilizes physics-based ray tracing to accurately simulate light-material interactions, including shadows and reflections. This setup enabled the automatic generation of a dataset comprising 1,200 images with each image annotated with labels indicating rebar intersection points. A few example images are presented in Figures 8. The dataset was subsequently divided into training and testing sets with an 80:20 ratio, resulting in 960 training samples and 240 testing samples.

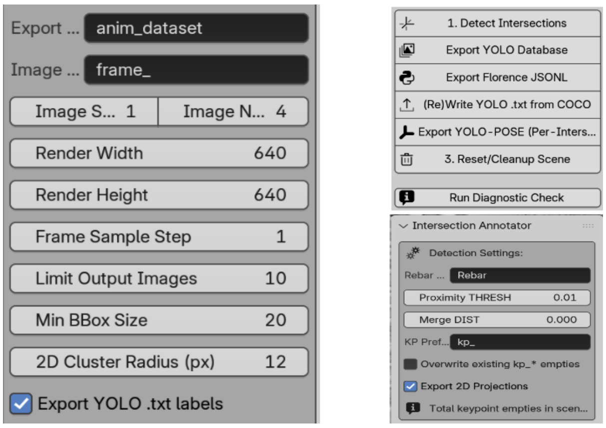

Figure 5. Screenshots of configurable export module for automated synthetic dataset generation

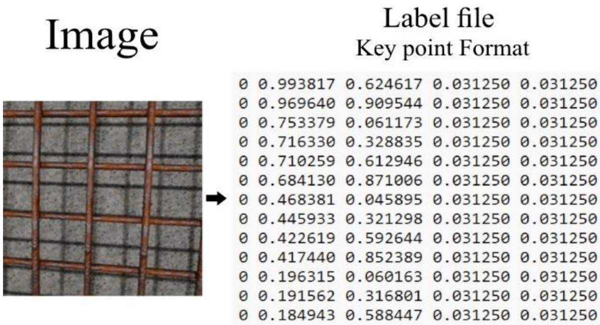

Figure 6. Example of a synthetic rebar mesh image with key-point annotation

2.2 Rebar intersection detection model training

The training configurations for the rebar intersection key-point estimation models adheres to the default settings of the UY framework. The training process utilizes images resized to 640×640 pixels. The models underwent training for 100 epochs with a batch size of 16, adjusted as needed to suit GPU memory limitations. Mosaic augmentation was activated during the initial stages of training and was disabled after 10 epochs to facilitate stabilization of convergence. These settings were chosen to strike a balance between training stability, localization accuracy, and computational efficiency.

We assess rebar intersection key-point estimation utilizing metrics proposed in the UY-Pose framework. The accuracy of key-point estimation is evaluated using keypoint-based metrics, such as mean Average Precision for keypoints (mAPkp).

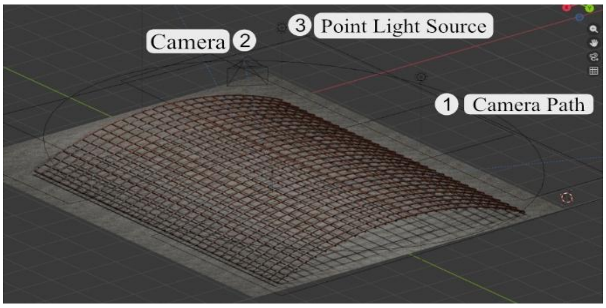

Figure 7. Camera and lighting configuration for multi-view rebar mesh image generation

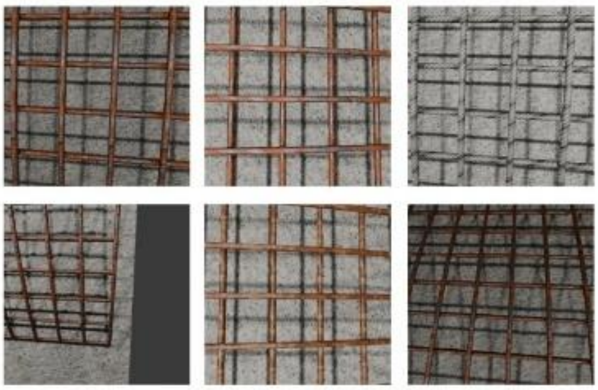

Figure 8. Examples of synthetic rebar mesh images showing appearance

2.3 Visual Servoing Experiment

2.3.1 Robot Simulation Environment

As illustrated in Figure 9, a robotic simulation environment was created using Webots to assess the proposed framework for identifying rebar intersection points and simulating visual servoing techniques. In this simulation was executed with a fixed controller sampling time 32 ms (31hz). A table measuring 1.0 m in width, 1.18 m in length, and 1.18 m in height was positioned with respect to the floor coordinate frame at (0.0 m, −2.11 m). A UR5 robotic manipulator was placed at (−0.138 m,

−1.35 m, 1.18 m). In addition, a parameterized 3D rebar mesh, with dimensions of 3.0194 m in width and 2.4 m in height representing the top-layer rebar mesh of a tunnel segment was modelled and placed within the simulation environment at coordinates (x: 2.8m, y: -1.5m m, z: 2.61 m). To provide visual feedback for both IBVS and PBVS experiments, an eye-in-hand camera system, which includes a co-located depth camera (Kinect RGB-D), was installed on the robot's end-effector. Trained vision models were integrated into the simulation to provide estimates for intersection points, which were then used as target points for IBVS and PBVS experiments.

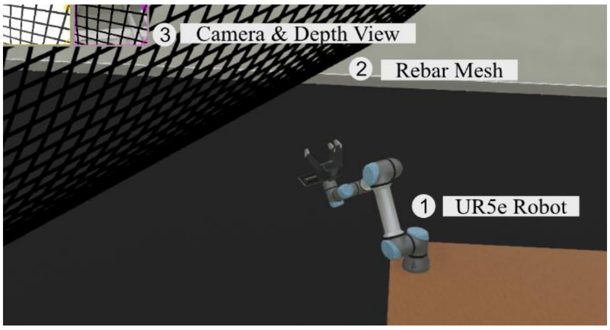

Figure 9. Simulation setup for experimenting rebar tying using vision-guided robotic system

2.3.2 Visual Servoing (IBVS and PBVS) simulation

In this experiment, IBVS and PVBS strategies are combined with YOLO-based (YB) rebar intersection point detection model. IBVS employs a Kanade–Lucas–Tomasi (KLT) tracker to follow a detected intersection point, and applies a control law based on the pseudo-inverse of the interaction matrix with gain scheduling to ensure stable convergence while minimizing the image-space error relative to the image center. PBVS uses the camera's intrinsic parameters together with RGB-D depth data to back-project clustered intersection points into 3D coordinates in the camera frame. Rather than relying on an explicit robot kinematic model, a numerically calibrated Jacobian maps joint motions to Cartesian displacements. By respecting practical kinematic constraints, the PBVS controller achieves accurate target alignment by minimizing the Euclidean 3D position error between the current and desired target locations. The effectiveness of the visual servoing strategies is assessed through both image-space and Cartesian metrics. For IBVS, convergence is determined by monitoring the decrease in pixel error between the actual and target intersection point locations in image space. In contrast, PBVS evaluates performance by calculating the Euclidean distance between the current and desired 3D positions. In both cases, accuracy and control behavior are compared based on the final positioning error. Several identified intersections are grouped together, arranged in row-major order, and visited one after the other.

3 Results

Figure 10 presents the evolution of the mAP achieved by the keypoint detection model on the testing set as the number of training epochs increases. The results demonstrate stable and effective learning on the proposed synthetic dataset, with no signs of significant overfitting. Furthermore, the YOLO key-point estimation model exhibits accuracy reaching a peak mAP of 98.5%, as shown in its Precision–Recall curve (P-RC) presented in Figure 11. The qualitative results shown in Figure 12 demonstrate that the key point detection model effectively and accurately detects rebar intersection points in the test images. This capability underscores the reliability and precision of the model in practical applications. These results confirm the models' ability to reliably identify rebar intersection points, even in unstructured construction environments.

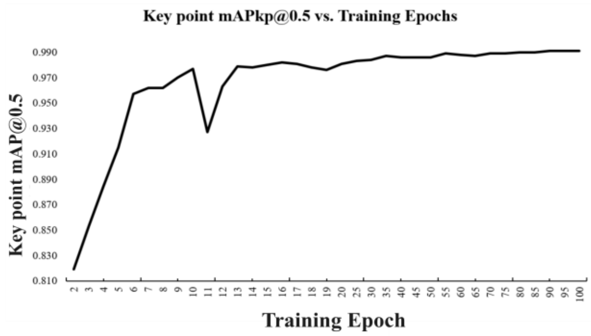

Figure 10. Training progress of the rebar intersection detection model, illustrating testing mAP@0.5[25]

Figure 13 and 14 shows the results of the IBVS and PBVS simulations, which were tested in the webots simulation environment using detected rebar intersection points as visual features. After detecting and clustering four intersection points, the IBVS and PBVS controllers successfully converged to each target sequentially. Figure 13 illustrates representative IBVS convergence results for two target intersection points (T1 and T2). As summarized in Table 1, four test points were evaluated. The initial image-space errors, ranging from 36.37 to 87.16 pixels, were reduced to below 3 pixels, resulting in accurate centering in the image plane. Convergence was achieved within 29 to 89 iterations, and the corresponding lateral Cartesian errors were reduced to sub-millimeter levels. These results demonstrate the robustness and stability of the IBVS strategy for robot alignment with the rebar intersection points.

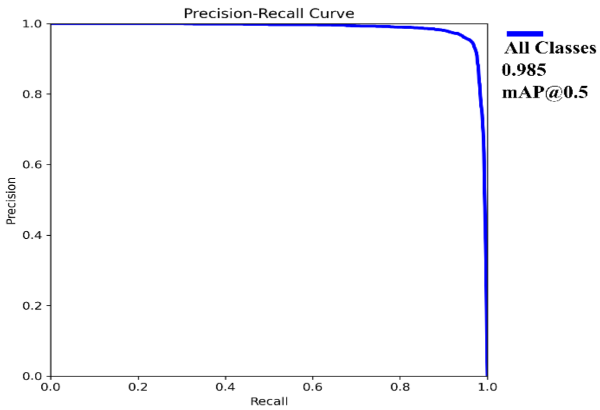

Figure 11. P-RC for rebar intersection key-point estimation (mAP@0.5 = 0.985)

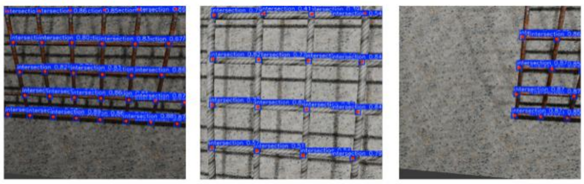

Figure 12. Examples of YOLO-based rebar intersection key-point detection results on test images.

Figure 14 shows the results of the PBVS approach for targets T3 and T4. As summarized in Table 2, the PBVS controller achieved sequential convergence to all targets, reducing lateral errors from initial offsets of 10.18–22.95 cm to below 0.5 cm, corresponding to error reductions of up to 98%. Convergence was reached within 46 to 86 iterations. Although depth control was limited due to the absence of independent camera translation along the optical axis, the PBVS approach demonstrated accurate and stable lateral alignment under realistic kinematic constraints.

Table 1. IBVS Convergence results for four rebar intersection targets

Target Intersection Points	Initial Error(px)	Final Error (px)	Iteration to converge
T1	36.37	2.87	29
T2	87.16	2.49	89
T3	51.92	2.65	70
T4	86.83	2.67	80

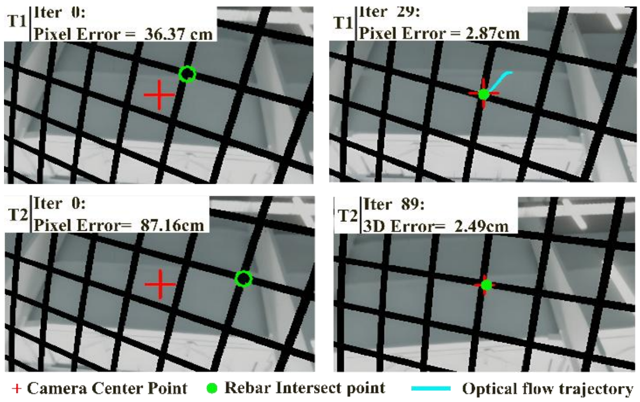

Figure 13. IBVS image-space convergence for rebar intersection points: (a) initial feature positions, (b) converged state for T1, and (c) converged state for T2

Table 2. PBVS convergence performance for four rebar intersection targets

Target Intersection Points	Initial Error(cm)	Final Error (cm)	Iteration to converge
T1	10.18	0.49	86
T2	22.96	0.47	62
T3	16.74	0.36	46
T4	23.54	0.49	59

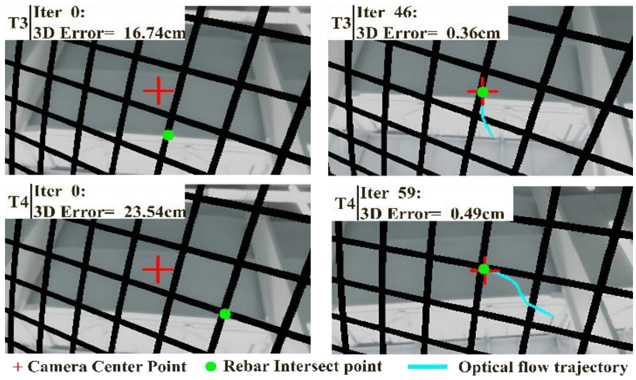

Figure 14. PBVS image-space convergence for rebar intersection points: (a) converged state for T3, and (b) converged state for T4

4 Discussion

Leveraging synthetic-data-trained key point detection models, this study demonstrates that IBVS and PBVS effectively support vision-guided robotic alignment with rebar intersection points. The experimental results confirm that the proposed synthetic data generation pipeline produces reliable visual and geometric representations of rebar meshes, allowing deep learning models to accurately learn and recognize rebar intersection points. The trained vision model was seamlessly integrated into robotic control pipelines, where they provided robust estimates of intersection points for both IBVS and PBVS control strategies. The simulation results affirm that the identification of rebar intersection points can be effectively coupled with visual servoing approaches, underscoring the feasibility of applying this framework to automated rebar tying operations in construction environments. The developed Blender plugins enable extensive randomization of geometry, appearance, lighting, and camera viewpoints, thereby eliminating the need for manual data collection and annotation. This not only significantly reduces the cost of dataset creation but also enhances data diversity and model generalization. Ultimately, the proposed synthetic data generation to robot simulation framework lays the groundwork for future developments. Notably, the generated synthetic datasets and simulation environment can be integrated with advanced physical simulation platforms to train reinforcement learning policies for dexterous manipulation tasks related to rebar intersection detection and tying. These extensions represent a promising avenue toward fully autonomous robotic systems for rebar assembly. This preliminary exploratory study has several limitations that highlight significant avenues for future research. Firstly, while the trained vision models exhibit strong performance in simulated environments, they have yet to be validated in real-world rebar mesh applications. Although the synthetic dataset demonstrates efficacy under controlled conditions, its generalizability to diverse real-world scenarios marked by varying lighting conditions, occlusions, and material properties. Future investigations should include more diverse 3D rebar configurations and real-world variability in the synthetic data generation pipeline, as well as expanded visual servoing trials with randomized initial conditions and real world hardware testing on physical rebar meshes. Secondly, current vision models are limited to detecting intersections formed by pairs of rebars within single-layer meshes. Scaling effectively to multi-layer setups would require retraining a 3D information-annotated model for multi-layer configurations and implementing occlusion-aware target selection strategies. This represents a key research opportunity for complex reinforcement layouts in large-scale construction projects. Thirdly, varying rebar mesh orientations and spatial configurations directly affect the vision model's ability to accurately identify intersection points, subsequently degrading both IBVS and PBVS controller convergence stability. Although convergence is achieved in the experiments presented, the optimal orientation and positioning of curved rebar mesh within the robot workspace markedly impact servoing stability and convergence speed. Identifying the best rebar mesh configurations and camera viewpoints for reliable IBVS and PBVS remains an unresolved research issue. Furthermore, the majority of existing studies on robotic rebar tying have primarily concentrated on planar horizontal or vertical rebar arrangements, thus leaving a critical gap in addressing the handling of curved rebar mesh intersections. Developing robust robotic path planning and control strategies for curved rebar meshes using IBVS and PBVS constitutes a vital direction for future research. Finally, while this study demonstrates the effectiveness of synthetic data and automated annotation tools in identifying rebar intersection points, there is a pressing need for a comprehensive study to assess the impact of synthetic-to-real domain gaps.

5 Conclusion

This study proposes an end-to-end framework for synthetic data generation and perception model evaluation for rebar intersection detection in automatic rebar tying using a vision-guided robotic system. The framework integrates parametric modeling, physics-based rendering, deep learning–based rebar intersection detection, and visual servoing simulation for robotic alignment. Perception models trained exclusively on synthetic data are deployed in a robotic simulation environment and evaluated using both IBVS and PBVS with a camera-mounted setup. The YB keypoint model trained on the generated synthetic dataset achieved a mAP of 98.5% for rebar intersection detection. To evaluate IBVS and PBVS, the trained models were into a robotic simulation. PBVS demonstrated strong alignment in Cartesian coordinates, with lateral errors reduced to less than 0.5 cm, even under depth constraints. Meanwhile, IBVS achieved stable convergence, with image errors reduced to below 3 pixels and lateral accuracy within sub-millimeter range. These results validate the effectiveness of the proposed end-to-end framework, demonstrating that integrating visual servoing with synthetic data-driven perception, provides a scalable and reliable foundation for automated rebar tying in complex construction environments.

Acknowledgement

We sincerely thank the Centre of Excellence on Technologies for Low Carbon and Lean Construction (TLC2), IIT Madras for funding this research, and our industry partners for their valuable support in providing access to real-world data.

References

[1] *World population prospects 2019: highlights*. New York: United Nations, 2019.

[2] J. K. Brueckner, "Urban Sprawl: Diagnosis and Remedies," *International Regional Science Review*, vol. 23, no. 2, pp. 160–171, Apr. 2000, doi: 10.1177/016001700761012710.

[3] S. Thacker *et al.*, "Infrastructure for sustainable development," *Nat Sustain*, vol. 2, no. 4, pp. 324–331, Apr. 2019, doi: 10.1038/s41893-019-0256-8.

[4] S. Abbas and M. Nehdi, "Mechanical Behavior of Ultrahigh-Performance Concrete Tunnel Lining Segments," *Materials*, vol. 14, no. 9, p. 2378, May 2021, doi: 10.3390/ma14092378.

[5] C. A. Issa and A. Nasr, "An experimental study of welded splices of reinforcing bars," *Building and Environment*, vol. 41, no. 10, pp. 1394–1405, Oct. 2006, doi: 10.1016/j.buildenv.2005.05.025.

[6] Z. Wang, Y. Xiong, R. Horowitz, Y. Wang, and Y. Han, "Hybrid Perception and Equivariant Diffusion for Robust Multi-Node Rebar Tying," 2025, *arXiv*. doi: 10.48550/ARXIV.2509.00065.

[7] S. Afshari, H. Tourajizadeh, and V. Boomeri, "Development of a new automatic rebar tying robot and its adaptive force control," *Lat. Am. Appl. Res.*, vol. 55, no. 4, pp. 445–462, Oct. 2025, doi: 10.52292/j.laar.2025.3649.

[8] R. Feng, Y. Jia, T. Wang, and H. Gan, "Research on the System Design and Target Recognition Method of the Rebar-Tying Robot," *Buildings*, vol. 14, no. 3, p. 838, Mar. 2024, doi: 10.3390/buildings14030838.

[9] J. Jin *et al.*, "Robotic binding of rebar based on active perception and planning," *Automation in Construction*, vol. 132, p. 103939, Dec. 2021, doi: 10.1016/j.autcon.2021.103939.

[10] C. Li, W. Zhang, F. Li, M. Guo, and S. Fan, "Vision-Based Detection, Localization, and Optimized Path Planning for Rebar Intersections in Automated Construction," *Applied Sciences*, vol. 15, no. 13, p. 7186, Jun. 2025, doi: 10.3390/app15137186.

[11] M. Liu, J. Guo, L. Deng, S. Wang, and H. Wang, "Enhanced vision-based 6-DoF pose estimation for robotic rebar tying," *Automation in Construction*, vol. 171, p. 105999, Mar. 2025, doi: 10.1016/j.autcon.2025.105999.

[12] D. Zhang, W. Zhang, F. Li, K. Liang, and Y. Yang, "PNANet: Probabilistic Two-Stage Detector Using Pyramid Non-Local Attention," *Sensors*, vol. 23, no. 10, p. 4938, May 2023, doi: 10.3390/s23104938.

[13] S. Chen, T. Shu, H. Zhao, G. Zhong, and X. Chen, "*TempEE*: Temporal–Spatial Parallel Transformer for Radar Echo Extrapolation Beyond Autoregression," *IEEE Trans. Geosci. Remote Sensing*, vol. 61, pp. 1–14, 2023, doi: 10.1109/TGRS.2023.3311510.

[14] M. Jagersand, O. Fuentes, and R. Nelson, "Experimental evaluation of uncalibrated visual servoing for precision manipulation," in *Proceedings of International Conference on Robotics and Automation*, Albuquerque, NM, USA: IEEE, 1997, pp. 2874–2880. doi: 10.1109/ROBOT.1997.606723.

[15] "A review on vision-based control of robot manipulators," *Advanced Robotics*, vol. 17, no. 10, pp. 969–991, Jan. 2003, doi: 10.1163/156855303322554382.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA: IEEE, Jun. 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.

[17] A. Kendall, M. Grimes, and R. Cipolla, "PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile: IEEE, Dec. 2015, pp. 2938–2946. doi: 10.1109/ICCV.2015.336.

[18] S. Mahendran, H. Ali, and R. Vidal, "3D Pose Regression using Convolutional Neural Networks," 2017, *arXiv*. doi: 10.48550/ARXIV.1708.05628.

[19] C. Yu, Z. Cai, H. Pham, and Q.-C. Pham, "Siamese Convolutional Neural Network for Sub-millimeter-accurate Camera Pose Estimation and Visual Servoing," 2019, *arXiv*. doi: 10.48550/ARXIV.1903.04713.

[20] T.-W. Huang, Y.-H. Chen, J. J. Lin, and C.-S. Chen, "Deep learning without human labeling for on-site rebar instance segmentation using synthetic BIM data and domain adaptation," *Automation in Construction*, vol. 171, p. 105953, Mar. 2025, doi: 10.1016/j.autcon.2024.105953.

[21] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," 2014, *arXiv*. doi: 10.48550/ARXIV.1405.0312.

[22] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Apr. 10, 2015, *arXiv*: arXiv:1409.1556. doi: 10.48550/arXiv.1409.1556.

[23] Y. Gao and K. M. Mosalam, "PEER Hub ImageNet: A Large-Scale Multiattribute Benchmark Data Set of Structural Images," *J. Struct. Eng.*, vol. 146, no. 10, p. 04020198, Oct. 2020, doi: 10.1061/(ASCE)ST.1943-541X.0002745.

[24] O. Michel, "Webots: Professional Mobile Robot Simulation," *Journal of Advanced Robotics Systems*, vol. 1, no. 1, pp. 39–42, 2004.

[25] A. Pal and S.-H. Hsieh, "Deep-learning-based visual data analytics for smart construction management," *Automation in Construction*, vol. 131, p. 103892, Nov. 2021, doi: 10.1016/j.autcon.2021.103892.