EPiC Series in Built Environment

Volume 6, 2025, Pages 482–491

Proceedings of Associated Schools of Construction 61st Annual International Conference

A BIM-Integrated Painting Approach in Construction with Autonomous Painting Robots

Chinedu Okonkwo[1], Amanda Betancourt[1], Sandeep Langar[1], and Ibukun Awolusi[1]
[1]The University of Texas at San Antonio

Integrating robotics into construction is gaining momentum as a viable solution to industry challenges, such as low productivity, on-site errors, and others. Among these challenges, painting activity stands out due to its inherent hazards, including exposure to harmful substances and falls. Autonomous painting robots (APRs) emerge as a promising means to mitigate these risks. Although these APRs are equipped with sensing technologies to facilitate their task execution, the dynamic nature of construction environments often leads to errors generated by onboard sensors. This study presents an innovative approach that leverages Building Information Modeling (BIM) to enhance painting accuracy and efficiency. By extracting semantic and geometric data from BIM models, the system improves sensor data interpretation, enabling precise identification of building elements and boundary delineation for painting tasks. Furthermore, integrating project scheduling ensures collision avoidance with concurrent construction activities, enhancing overall task coordination and efficiency. This study addresses sensor-related errors inherent in construction robotics, offering a comprehensive solution for safer, more efficient painting operations.

Keywords: Automation; Autonomous Painting Robots; Building Information Modeling; Construction Robotics.

Introduction

The adoption and implementation of robotics on construction sites have the potential to improve efficiency and safety significantly (Xu & deSoto, 2020). Of the various robotic systems that can be adopted and implemented on construction sites, autonomous painting robot (APR) can execute painting tasks with limited supervision and generate significant benefits to the potential adopters, such as minimized safety risks and improved productivity, when compared to traditional painting methods (Naticchia et al., 2007). Workplace safety incidents such as falls from heights, respiratory problems from exposure to toxic fumes, and musculoskeletal disorders (Park et al., 2016) can be significantly reduced with the implementation of APR. Unlike traditional painting methods that rely mainly on the expertise of human painters, APR can be designed to execute painting jobs with higher accuracy and speed than human painters and can be customized to adapt to different specifications when needed (Kahane & Rosenfeld, 2004). Also, APR is unsusceptible to most human limitations such as fatigue, distractions, and others, presenting the possibility of continuous operation when needed, leading to increased productivity and timesaving, especially in large-scale and/or time-constrained projects.

APRs are equipped with different sensors, algorithms for data collection, and processing to enable task execution. Vision sensors consist of cameras, color sensors, and dynamic vision sensors and stand at the frontline of these sensors. RGB-D cameras on an APR can capture detailed spatial images (including the surface to be painted), enabling APR to navigate the work environment and make necessary adjustments to achieve the desired painting trajectory (Zhang et al., 2015; Okonkwo & Awolusi, 2025). The sophisticated algorithms onboard APR, powered by artificial intelligence (AI), allow them to analyze visual data and make intelligent decisions (Asadi et al., 2021). Some of these decisions include appropriate brush strokes or spray pressure based on the texture of the surface. Brush-based painting robots are equipped with tactile sensors to enhance interaction with the surface. These robots can maintain brush stroke consistency and ensure uniform surface painting by gauging the pressure exerted during painting (Asadi et al., 2018).

Since environmental factors such as temperature and humidity play a vital role in painting operations (Baglioni et al., 2016), APRs are equipped with sensors to monitor real-time environmental conditions (temperature and humidity). By monitoring real-time environmental conditions, APR can make necessary adjustments to the painting parameters to attain the desired painting quality with reduced risk of defects (such as blemishes, early flaking, and others). Despite the presence of these sophisticated sensors, their implementation in the construction industry is very limited. One of the reasons for such limited adoption/implementation can be attributed to the complexity of the work environment (Chen et al., 2018). The construction environment is dynamic, with co-occurring activities and the constant movement of workers, equipment, and materials. Further, changing lighting conditions can interfere with data acquisition by the APR onboard sensors, leading to errors and inconsistencies.

By providing a digital representation of the physical components of a facility, Building Information Modeling (BIM) can reduce the error rate of APRs (Gagliardo et al., 2024). BIM is a three-dimensional virtual model representing a facility's physical and functional characteristics (Ding et al., 2014). BIM facilitates dynamic planning and coordination by updating the digital model with any changes on the construction site. Relying on this information, APRs can adjust painting based on the most updated model, reducing the need for human intervention. To improve accuracy by leveraging the parametric digital models generated with BIM, APR can compare the sensor data with the generated digital models, thereby allowing better decision-making (Kahane & Rosenfeld, 2004). BIM can also standardize surface specifications in the digital model, which can guide the development of the program used by APRs to adapt painting techniques (de Lima et al., 2023).

Background

BIM is a modeling process that depicts multi-dimensional models of projects in the construction industry (Tang et al., 2019). It is an elaborate data management tool consisting of datasets pertinent to building construction (such as schedule, cost management, and others) (Ding et al., 2014) and provides stakeholders (such as designers, contractors, and others) the opportunity to generate discipline-specific usable information (Langar & Pearce, 2014; Langar & Pearce, 2017: Fountain & Langar, 2018) depending on the maturity of the users (Fountain & Langar, 2018). BIM provides stakeholders enhanced transparency through comprehensive visual models with project specifications (Azhar, 2011). These essential details can impact decision-making, allowing access to problem mitigation and identifying alternatives (Rock et al., 2018). Data is integrated into diverse disciplines, enabling the creation of detailed digital representations of the input data sets.

The Industry Foundation Curriculum (IFC) serves as a standardized digital description of the built environment and is a standardization for data incorporated within BIM design. IFC can be used across diverse hardware, software, and numerous application interfaces, allowing information exchange between construction stakeholders and archiving project information (IFC 2024). The IFC schema used in BIM contains over 600 data categories (Liu et al., 2021). Visual elements and identifier codes are subject to variation based on the specific schema version and the category under which they fall. These visual IFC codes represent various elements, including walls, doors, windows, floors, roofs, and others.

Construction robots have shown capabilities in performing tasks traditionally completed by the skilled labor force. According to Weaver (2018), as cited in Kim et al. (2021), a bricklaying robot by Construction Robotics increased the bricklaying speed by 3–5 times. The Demonstration Learning Study experiments showed that construction robots can be programmed to perform repetitive work tasks in a geometrically adaptive environment, executing work while sensing these changes (Liang et al., 2019). According to Shaw (2015), DPR Construction has launched a robot specifically designed to improve the efficiency of creating drywall layouts. Thus, construction robots can effectively reduce the number of work hours by performing tasks at a higher efficiency and eliminating risks and limitations associated with the traditional workforce.

APR currently uses different sensors such as Light Detection and Ranging (LiDAR), ultrasonic, and cameras to collect geometric data and used for creating 3D mapping of construction site zones for painting (Asadi et al., 2020; Frintrop et al., 2005). The 3D data collected from these sensors are not always accurate due to incomplete data sets and poor visual data. According to Ibrahim et al. (2019), challenges in the current methods of using LiDAR point cloud data in conjunction with BIM models include: inefficient manual data collection planning, lack of visual in-progress feedback, inaccuracies in the data collection plan and execution, loss of time, and costly manual placement of visual tags; in addition to high investment costs for specialists and the required equipment. These challenges currently faced with the existing methods establish the need for a new approach in developing a BIM integration framework for APR to improve efficiency and productivity. Current APR functions primarily through the use of LiDAR and Radio Detection and Ranging (RADAR); BIM could serve as an augmentation system for sensor data, identifying variances in the sensor's datum data, and the specified dimensions of the surveyed zone, mitigating errors and improving accuracy.

Most studies on BIM integration in autonomous robots in construction have focused on navigation and path planning, with limited studies on construction task execution, such as painting (Chen et al., 2022; Hamieh et al., 2020; Zhao & Cheah, 2023). Gagliardo et al. (2024) proposed a BIM-integrated robotic color spraying approach using a fixed robotic arm. While this approach utilized semantic and geometric information from BIM, its reliance on manual efforts to position the robotic arm correctly limits its applicability in real-world scenarios and increases its susceptibility to human errors. Therefore, to address these limitations, this study presents a BIM-integrated APR framework that significantly reduces task execution errors by leveraging onboard sensors for autonomous operation, thereby minimizing the need for manual human intervention.

Research Method

To achieve the study's objective, a scoping review was conducted, sourcing relevant literature from the Scopus and Web of Science databases. Initially, 104 articles were identified, but a preliminary review led to the exclusion of 57 articles, primarily due to duplication or misalignment with the study's focus. The selection criteria included peer-reviewed articles and conference papers published in English, specifically examining BIM integration in autonomous robotic task planning and execution within the construction industry. After a full-text evaluation, an additional 22 studies were removed, narrowing the final sample to 25 (Figure 1). These selected studies were then critically analyzed, and findings were synthesized to inform the development of a BIM integration framework tailored for autonomous robotic applications in construction.

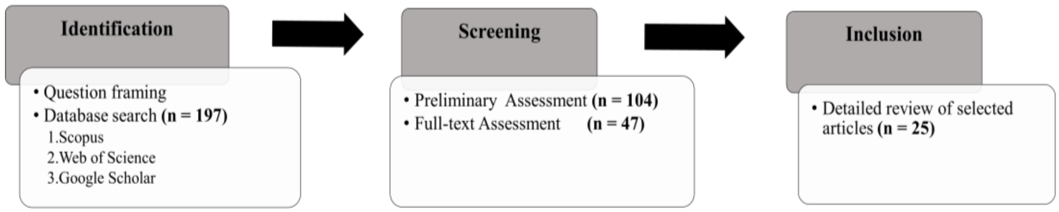

Figure 1. Research Methodology

Results

The framework's structure and phases—data extraction, analysis, task planning, and execution—were developed based on the findings from the review of existing literature on BIM integration in autonomous mobile robots (Figure 2). These phases align with established methodologies in robotic systems and BIM-based task optimization (Pan and Zhang, 2021). *Data extraction* (first stage) involves identifying relevant building information necessary for painting activities and extracting semantic and geometric data. Onboard sensors such as cameras and LiDAR are utilized to gather sensory data, while the project schedule is incorporated to aid in project planning. BIM data serves as a reference to enhance the accuracy of onboard robot sensors, thereby reducing errors. In the second stage (*Data Analysis*), APR uses the acquired sensory data to segment various building elements, preparing for painting tasks. BIM data is crucial in verifying the accuracy of surface boundaries generated by the APR. By providing a specified error margin, APR can make necessary adjustments to incorrect boundary identifications, ensuring precision in painting. The third stage (*Task Planning*) involves generating painting plans by APR based on information obtained during the previous phase. This includes determining the spraying angle and distance from the surface for optimized painting efficiency and quality. In the fourth phase (*Task Execution*), APR executes the painting task, implementing plans generated in the previous stages.

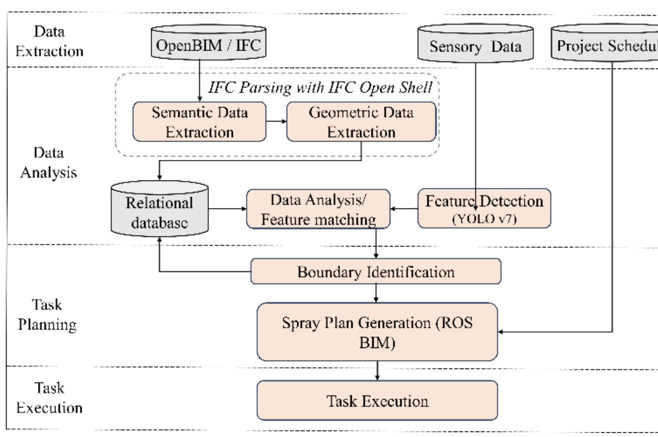

Figure 2. BIM integration framework for autonomous wall painting

Data Extraction

The framework consists of three data sources: IFC, Sensory data from onboard sensors, and the project schedule. IFC standards serve as a common data model for exchanging information between different software applications. Five IFC elements considered in this framework are: IfcWall (wall information), IfcDoor (door details), IfcWindow (windows properties), IfcOpeninigElement (any wall opening parameter(s) other than doors and windows), and IfcCovering (ceiling properties) (Figure 3).

Industry Foundation Curriculum (IFC)

IfcWall represents vertical building elements that define the boundaries of spaces. For APR applications, IfcWall data serves as a foundation for understanding the layout of surfaces requiring painting (IfcWall, 2024). By analyzing wall properties such as dimensions, orientation, and surface material, APR can generate optimized painting plans, adjust nozzle angles, and regulate paint flow rates to achieve uniform coverage across walls. IfcDoor provides information on openings within building elements that allow passage. By extracting data related to IfcDoor from BIM models, robotic painting systems can intelligently navigate construction sites, identify locations, and adjust painting patterns accordingly. Additionally, door dimensions and materials information can inform robotic painting parameters, ensuring precise coverage, and minimizing wastage (IfcDoor, n.d.). IfcWindow represents openings within building elements that provide views, daylight, or ventilation. Windows introduces complexity to APR tasks due to their varied shapes, sizes, and configurations. Integrating IfcWindow data into APR enables precise identification and characterization of window surfaces. APR can adapt painting strategies to accommodate windows, including masking off areas, adjusting spray patterns, and controlling paint overspray. Leveraging IfcWindow information enhances the accuracy and efficiency of painting processes while minimizing the risk of paint damage to window components. Similar to three elements, IfcOpeningElements and IfcCovering provide information on voids and coverings/claddings. Information from these elements is extracted in the data analysis stage.

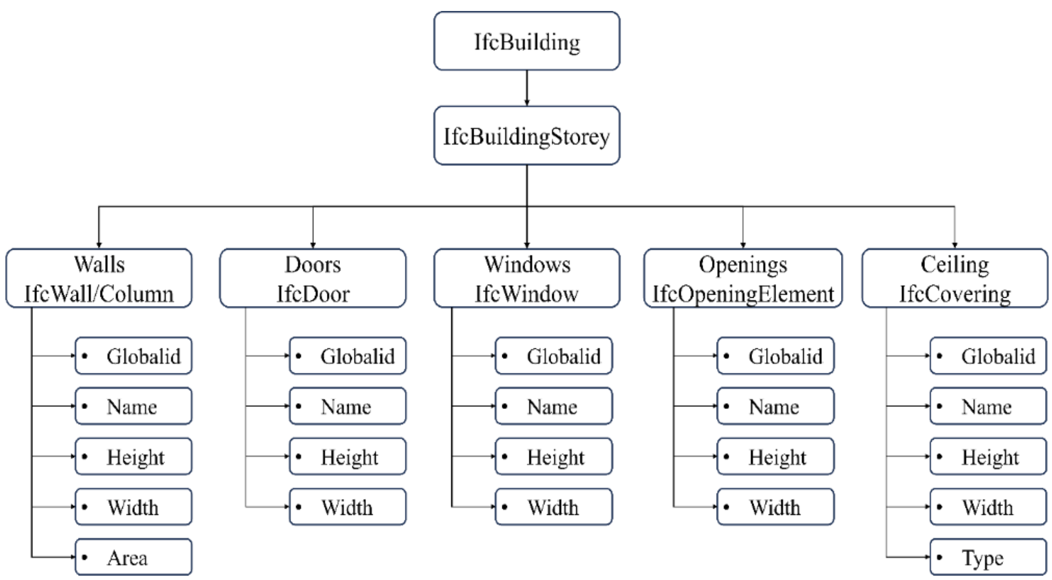

Figure 3. IFC element parameters

Sensory Data

The sensory data considered in this study are images from an onboard camera on the APR. Although there are various devices for environmental sensing, only a camera was considered in this framework to simplify the process by avoiding the complexity of multiple data analyses. Also, since this study is concerned primarily with painting task execution and not path planning of the mobile robot, images from a camera would suffice for object identification and segmentation. Images from onboard cameras on an APR provide real-time visual information on the work environment. These images are then processed with computer vision algorithms to extract relevant features like edges and contours. The real-time images from cameras facilitate quality control, ensuring uniform paint application and prompt detection of defects (Asadi et al., 2021). Despite the above advantage of camera images, real-time images on a construction site can be affected by environmental factors like lighting conditions, occlusion, and shadows, which negatively affect the accuracy and quality of building element identification. By incorporating BIM data from IFC, these errors can be reduced by augmenting the feature, extracting data from onboard cameras with BIM data.

Project Schedule

Project schedule integration in robotic painting improves task coordination by aligning the task planning with the overall project schedule to minimize conflicts with other construction tasks. By continuously monitoring updates in the project schedule, APRs can adjust their operation to accommodate developing project requirements.

Data Analysis

In this phase, IFC data and sensory data obtained from the onboard sensor (camera) are analyzed. IFC data are parsed with IfcOpenShell to extract semantic and geometric information. Simultaneously, images from onboard cameras are analyzed with YOLO v7 for feature identification and segmentation. Results from the feature identification are compared with IFC data to improve prediction accuracy and reduce segmentation errors. The relational database in this phase is a storehouse of boundary information from IFC information and final information based on both IFC and sensory data. Juxtaposing this information will provide insight into the level of accuracy of project execution, as well as provide facility managers with as-built dimensions of the project.

IFC Data Parsing

IFC data parsing aims to extract geometric and semantic information from the BIM model. Geometric information describes an element's shape, size, and position in a 3D environment (Xu et al., 2022). It also includes lines, curves, and volumes that collectively define the physical form of a building element. Semantic information includes element type, material properties, function, and other non-geometric data that provide contextual meaning to a building element (Karan et al., 2015). In this study, IfcOpenShell is employed for IFC data parsing. IfcOpenShell is an open-source library for parsing IFC file format (IfcOpenShell, 2024) and pseudocode for implementing IfcOpenShell (Figure 4).

```
Import libraries and dependencies
  IfcOpenShell
  Other data manipulation libraries
Define task functions like:
  Loading an IFC file
  Accessing and manipulating data in the IFC file
  Information extraction from the IFC file
Main program:
  // Load IFC file
  ifc_file_path = "specify path"
  ifc_model = IfcOpenShell.Open(ifc_file_path)
  // Access information from the IFC model
  entities = ifc_model.GetEntities()
  // Example: Extract specific information (e.g., doors)
  walls = ifc_model.GetDoors()
  for door in doors:
     door_properties = wall.GetProperties()
     print("Door:", door_properties)
  // Other operations like geometry processing, visualization, etc.
Handle errors gracefully and close the IFC model properly
  Catch exceptions and errors
  Close the IFC model and release resources
```

Figure 4. Pseudocode for implementing IfcOpenShell adopted in the study

Feature Detection

YOLO v7 is the feature detection algorithm adopted in this framework. The YOLO v7 model, characterized by its single-pass architecture and efficient object localization (Wang et al., 2022), proves instrumental in feature detection for task planning in painting robots. YOLO v7 facilitates semantic understanding of the environment, allowing APR to differentiate between different surfaces and apply appropriate painting techniques. First, annotated images of the painting environment are collected and labeled with bounding boxes around relevant features. The dataset should include diverse scenarios to ensure the model's generalization capability. This is followed by training the algorithm with the annotated dataset to learn the visual representations of features. In situations like insufficient training data, transfer learning techniques can be employed by fine-tuning pre-trained models on large-scale datasets like ImageNet to improve performance (Krizhevsky et al., 2017). Upon completion of the algorithm training, it is integrated into the APR operating system. This integration involves developing interfaces for image acquisition from onboard cameras, real-time inference using the trained model, and decision-making based on detected features.

Feature Matching

BIM data are converted into YOLO v7 compatible format to integrate with camera image data. For this study, 3D point cloud is adopted as the preferred format. A 3D point cloud is a group of data points representing measurements of an object's surface (Guo et al., 2021). Figure 5 below provides an overview of the data analysis and feature mapping steps.

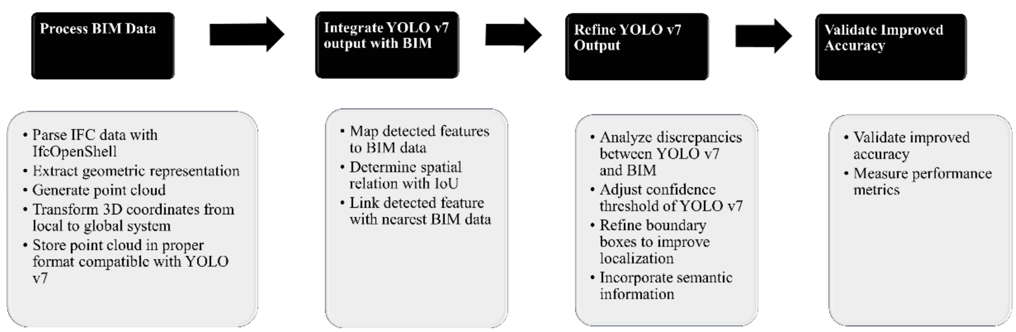

Figure 5. Data analysis and feature-matching steps

To convert BIM data to point cloud, the IFC format is parsed with IfcOpenShell to extract geometric and semantic information about the building. The building elements' geometric representation, such as boundary representation, are converted to geometric primitives like edges and vertices, which are more suitable for point cloud generation. Subsequently, the sampling strategy to generate points representing the surface of building elements is defined. For instance, planar surfaces like walls and floors are sampled within the surface boundaries, while points that approximate the surface geometry accurately are generated for curved surfaces by adaptive sampling. To capture detailed geometry without compromising efficiency, point density and distribution are considered in the process. The generated 3D point coordinates are converted from the local IFC file system to the global coordinate system to ensure consistency between the coordinate system of the point cloud and other data used in the system optimization. Detected features, such as bounding boxes from YOLO v7, are mapped to the corresponding BIM model elements. The spatial relationship between the image prediction and the BIM model is analyzed by the Intersection over Union (IoU) technique. IoU is a metric used to measure the overlap between two bounding boxes, in this case, the prediction from YOLO v7 and that of the BIM elements. IoU operates by calculating the area of intersection and area of union between the bounding boxes and computes the ratio of intersection area to union area (Rahman & Wang, 2016). Based on the ratio, the accuracy of the prediction of YOLO v7 is determined. Discrepancies between the BIM model and image predictions are analyzed to identify necessary adjustments to the confidence threshold of the YOLO v7 algorithm. At this stage, semantic information from the BIM model, such as material type and surface properties, are incorporated to improve the prediction quality. Finally, the improved accuracy of the model is validated by comparing the refined predictions with the BIM model.

Task Planning

To plan a painting task, boundaries of building elements are established based on the data analysis and feature mapping obtained from YOLO v7 feature detection and parsed IFC data. Before task planning, final boundary identification information is stored in the relational database to provide building as-built dimensions, which facility managers can use for building maintenance. The data from defined boundaries and building elements is sent to the Robot Operating System (ROS) node, where the spraying motion plan is generated. ROS is an open-source framework designed to facilitate robotic software development (Iñigo-Blasco et al., 2012). The modularity and hardware abstraction features of ROS allow developers to program the functioning of the nodes to communicate with each other, enabling seamless integration of different functionalities (Hellmund et al., 2016), with the basic ROS concept operation depicted in Figure 6.

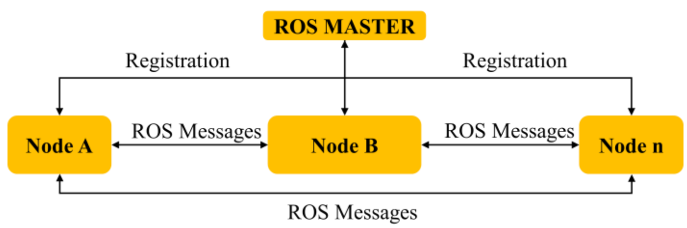

Figure 6. Robot Operating System (ROS) operation concept (source: clearoathrobotics.com)

Task Execution

The execution of the painting task relies on ROS's motion control and coordination capabilities to translate planned actions into physical movements of the APR. ROS nodes control the robot's actuators, such as motors, to drive APR movement and adjust painting tools (Iñigo-Blasco et al., 2012). ROS nodes are pivotal in controlling the APR's actuators, including motors responsible for driving the robot's movement and motors for adjusting painting tools such as spray nozzles. These nodes communicate with hardware drivers and interface with the robot's control system, issuing commands to regulate velocity, orientation, and actuator positions with high accuracy and reliability. Additionally, ROS provides libraries for implementing feedback control loops, enabling APR to continuously monitor its state and adjust its actions in response to environmental changes or deviations from the planned trajectory. Project schedule information is integrated into the task planning process with ROS to improve the APR's coordination within the project. With access to real-time project timelines, APR can prioritize tasks based on project constraints and optimize activities to align with project objectives.

By establishing communication between the APR and the ROS incorporated with BIM, the APR can receive calculated instructions and detailed painting parameters from the ROS system. These transmitted instructions can include the sequence for painting, specified quality standards, and surfaces to be targeted for task execution. During the painting process, the APR can transmit feedback back to the ROS to provide insightful data such as the project progression, status updates, and quality results. This communication bridges the software together to work cohesively to prioritize tasks, plan a detailed work schedule around conflicting work schedules in the same zone, determine time-saving painting paths, and use ideal paint spray settings to achieve quality results in a time-efficient manner.

Conclusion

This paper presents a conceptual framework for integrating BIM with APR to enhance productivity and reduce errors in task execution using a literature review. First, BIM data are parsed to extract semantic and geometric data to augment the sensory data from onboard sensors. For this study, YOLO v7 was adopted as the feature detection algorithm to process the images captured by onboard APR cameras. Subsequently, the BIM model and onboard camera visual data are juxtaposed to improve accuracy. The output is then fed to ROS for task planning and execution alongside data from the project schedule. In addition to addressing sensor-related errors in construction robotics, the framework provides facility managers with as-built information about the project, which is stored in a relational database.

Given the conceptual nature of this study, there are several limitations to this study. First, the framework's components are developed based on the findings of existing studies. Like most conceptual studies, the reliance on theory sometimes overlooks empirical realities and practical constraints that may arise in real-world scenarios. Further studies will look into the implementation of the framework and the feasibility and interoperability of the software proposed by this study. Also, knowledge of BIM is required to operationalize OpenBIM as part of the framework implementation.

References

Asadi, E., Li, B., & Chen, I.-M. (2018). Pictobot: A Cooperative Painting Robot for Interior Finishing of Industrial Developments. *IEEE Robotics and Automation Magazine*, 25(2), 82–94.

Asadi, K., Haritsa, V. R., Han, K., & Ore, J.-P. (2021). Automated Object Manipulation Using Vision-Based Mobile Robotic System for Construction Applications. *Journal of Computing in Civil Engineering*, 35(1), 04020058.

Asadi, K., Kalkunte Suresh, A., Ender, A., Gotad, S., Maniyar, S., Anand, S., Noghabaei, M., Han, K., Lobaton, E., & Wu, T. (2020). An integrated UGV-UAV system for construction site data collection. *Automation in Construction*, 112, 103068.

Azhar, S. (2011). Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AEC Industry. *Leadership and Management in Engineering*, 11(3).

Baglioni, P., Giorgi, R., & Chelazzi, D. (2016). The degradation of wall paintings and stone: Specific ion effects. Current Opinion in Colloid and Interface Science, 23, 66–71.

Chen, Q., García de Soto, B., & Adey, B. T. (2018). Construction automation: Research areas, industry concerns and suggestions for advancement. *Automation in Construction*, 94, 22–38.

Chen, Z., Chen, K., Song, C., Zhang, X., Cheng, J. C., & Li, D. (2022). Global path planning based on BIM and physics engine for UGVs in indoor environments. *Automation in Construction*, 139, 104263.

Ding, L., Zhou, Y., & Akinci, B. (2014). Building Information Modeling (BIM) application framework: The process of expanding from 3D to computable nD. *Automation in Construction*, 46, 82–93.

de Lima, P. R. B., de Souza Rodrigues, C., & Post, J. M. (2023). Integration of BIM and design for deconstruction to improve circular economy of buildings. *Journal of Building Engineering*, *80*, 108015.

Fountain, J., & Langar, S. (2018). Building Information Modeling (BIM) outsourcing among general contractors. *Automation in Construction*, 95, 107-117.

Frintrop, S., Rome, E., Nüchter, A., & Surmann, H. (2005). A Bimodal Laser-Based Attention System. *Computer Vision and Image Understanding*, 100(1), 124–151.

Gagliardo, A., Garbin, S., Terzer, M., Matt, D. T., & Giusti, A. (2024). A BIM-Integrated Robotics Application for Color Spraying in Construction. *In J. Fottner, K. Nübel, and D. Matt (Eds.), Construction Logistics, Equipment, and Robotics (pp. 194–200)*. Springer Nature Switzerland.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., & Bennamoun, M. (2021). Deep Learning for 3D Point Clouds: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12), 4338–4364.

Hamieh, A., Makhlouf, A. B., Louhichi, B., & Deneux, D. (2020). A BIM-based method to plan indoor paths. *Automation in Construction*, *113*, 103120.

Hellmund, A.-M., Wirges, S., Taş, Ö. Ş., Bandera, C., & Salscheider, N. O. (2016). Robot operating system: A modular software framework for automated driving. *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC),* 1564–1570.

IfcDoor. (n.d.). Retrieved March 31, 2024, from https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/schema/ifcsharedbldgelements/lexical/ifcdoor.htm

IfcOpenShell. (n.d.). *IfcOpenShell—IfcOpenShell 0.7.0 documentation*. Retrieved March 29, 2024, from https://blenderbim.org/docs-python/ifcopenshell.html

IfcWall. (n.d.). Retrieved March 31, 2024, from https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/TC1/HTML/ifcsharedbldgelements/lexical/ifcwall.htm

Iñigo-Blasco, P., Diaz-del-Rio, F., Romero-Ternero, M. C., Cagigas-Muñiz, D., & Vicente-Diaz, S. (2012). Robotics software frameworks for multi-agent robotic systems development. *Robotics and Autonomous Systems*, 60(6), 803–821.

Kahane, B., & Rosenfeld, Y. (2004). Balancing Human-and-Robot Integration in Building Tasks. *Computer-Aided Civil and Infrastructure Engineering*, 19(6), 393–410.

Karan, E., Irizarry, J., & Haymaker, J. (2015). Generating IFC models from heterogeneous data using semantic web. *Construction Innovation*, 15(2), 219–235.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84–90.

Langar, S., & Pearce, A. R. (2014). State of Adoption for Building Information Modeling (BIM) in the southeastern United States. *Proceedings of 50th Associated School of Construction International Conference*, Washington DC, USA, March 26-29.

Langar, S., & Pearce, A. R. (2017). Implementation trends for Rainwater-Harvesting Technologies and Strategies (RwHTS) and their Relationship Building Information Modeling (BIM). *ASCE Journal of Architecture Engineering*, 23 (1).

Naticchia, B., Giretti, A., & Carbonari, A. (2007). Set up of an Automated Multi-Colour System for Interior Wall Painting. *International Journal of Advanced Robotic Systems*, 4(4), 50.

Okonkwo, C., & Awolusi, I. (2025). Environmental sensing in autonomous construction robots: Applicable technologies and systems. *Automation in Construction*, *172*, 106075.

Park, H., Park, H. D., & Jang, J.-K. (2016). Exposure Characteristics of Construction Painters to Organic Solvents. *Safety and Health at Work*, 7(1), 63–71.

Pan, Y., & Zhang, L. (2021). A BIM-data mining integrated digital twin framework for advanced project management. *Automation in Construction*, *124*, 103564.

Rahman, M. A., & Wang, Y. (2016). Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation. *In G. Bebis, R. Boyle, B. Parvin, D. Koracin, F. Porikli, S. Skaff, A. Entezari, J. Min, D. Iwai, A. Sadagic, C. Scheidegger, and T. Isenberg (Eds.), Advances in Visual Computing (pp. 234–244)*. Springer International Publishing.

Rock, M, Hollberg, A., Habert, G., & Passer, A. (2018). LCA and BIM: Visualization of environmental potentials in building construction at early design stages. *Building and Environment*, 140

Tang, S., Shelden, D. R., Eastman, C. M., Pishdad-Bozorgi, P., & Gao, X. (2019). A review of building information modeling (BIM) and the internet of things (IoT) devices integration: Present status and future trends. *Automation in Construction*, 101, 127–139.

Wang, Y., Wang, H., & Xin, Z. (2022). Efficient Detection Model of Steel Strip Surface Defects Based on YOLO-V7. IEEE Access, 10, 133936–133944.

Xu, S., Wang, J., Wang, X., Wu, P., Shou, W., & Liu, C. (2022). A Parameter-Driven Method for Modeling Bridge Defects through IFC. *Journal of Computing in Civil Engineering*, 36(4), 04022015.

Zhang, L., Shen, P., Zhu, G., Wei, W., & Song, H. (2015). A Fast Robot Identification and Mapping Algorithm Based on Kinect Sensor. *Sensors*, 15(8).

Zhao, X., & Cheah, C. C. (2023). BIM-based indoor mobile robot initialization for construction automation using object detection. *Automation in Construction*, *146*, 104647.