



## **Machine Learning Enhanced Spatial-Data Workflow Redesign: A Quantitative Evaluation of Task Success and Operational Cost Reduction in Enterprise LiDAR Platforms**

**Md Ashrafuzzaman<sup>1</sup>; Gazi Yeasin Arifin<sup>2</sup>;**

[1]. IT Administrative Officer, Abdul Kadir Mollah City College, Bangladesh;  
Email: [md.ashrafuzzamanuk@gmail.com](mailto:md.ashrafuzzamanuk@gmail.com)

[2]. Kite Games Studio Ltd; Lead UX Designer, Dhaka, Bangladesh;  
B.S. Software Engineering, American International University Bangladesh, Dhaka, Bangladesh;  
Email: [arifin.yeasin@gmail.com](mailto:arifin.yeasin@gmail.com)

**Doi:** [10.63125/0den7c02](https://doi.org/10.63125/0den7c02)

**Received:** 19 March 2022; **Revised:** 25 April 2022; **Accepted:** 21 May 2022; **Published:** 08 June 2022

### **Abstract**

*This study quantitatively evaluated the effectiveness of machine learning (ML) enhanced spatial-data workflow redesign in improving task success and reducing operational costs within enterprise LiDAR platforms. A quantitative quasi-experimental comparative design was employed using 240 enterprise LiDAR processing tasks representing airborne, terrestrial, drone-based, mobile, and vehicle-mounted acquisition environments. Conventional and ML-enhanced workflows were compared across task success, first-pass completion, processing latency, analyst involvement, workflow reliability, computational resource utilization, reprocessing, and operational cost. Descriptive statistics, paired comparative tests, effect-size estimates, multivariable logistic regression, and regression-based cost and processing-time analyses were applied at a significance level of  $p < 0.05$ . The findings demonstrated that ML-enhanced workflow redesign increased task success from 78.3% to 92.5%, representing a 14.2-percentage-point improvement, while first-pass completion increased from 69.6% to 87.1%. Task failure decreased from 21.7% to 7.5%, and quality-control rejection declined from 18.8% to 6.7%. Mean total processing duration decreased by 33.0%, from 94.8 to 63.5 minutes per task, while turnaround time declined from 116.6 to 75.4 minutes. Analyst involvement decreased by 43.1%, from 37.6 to 21.4 minutes per task, and manual interventions declined by 45.2%. Reprocessing requirements decreased by 60.2%, additional retries by 62.0%, and rework duration by 56.0%. Mean operational cost declined from \$46.80 to \$32.10 per task, representing a 31.4% reduction, while cost per successful task decreased by 41.9%, from \$59.74 to \$34.70. Computational utilization showed a redistribution of resources, with CPU hours declining by 28.8% while GPU hours increased by 14.3%. Multivariable analysis showed that ML-enhanced processing remained independently associated with successful task completion (adjusted OR = 3.18, 95% CI [1.76, 5.74],  $p < 0.001$ ) after controlling for major dataset characteristics. The study established that ML-enhanced workflow redesign improved enterprise LiDAR performance through higher task success, shorter processing cycles, reduced human intervention, greater workflow reliability, and lower operational cost.*

### **Keywords**

*Machine Learning, LiDAR, Workflow Automation, Task Success, Cost Reduction.*

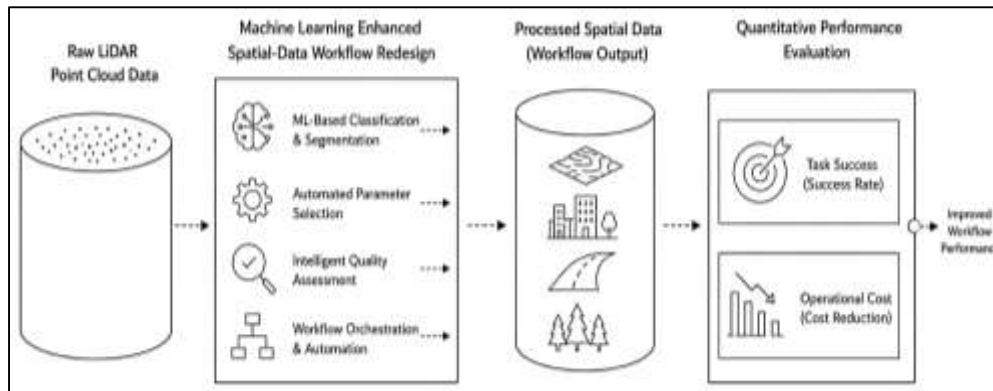
## INTRODUCTION

Light Detection and Ranging (LiDAR) is an active remote-sensing technology that determines the distance between a sensor and a target surface by transmitting laser pulses and measuring the time required for reflected energy to return to the sensor. The resulting measurements generate highly detailed three-dimensional representations of terrain, vegetation, buildings, roads, utilities, industrial facilities, and other physical environments (Marchesini & Farinelli, 2020). LiDAR observations are commonly organized as point clouds in which individual points contain three-dimensional coordinates and may also include intensity, return number, classification, scan angle, timestamp, color, and other sensor-derived attributes. Spatial data refers broadly to information associated with geographic positions, geometric properties, coordinate systems, spatial relationships, or measurable locations. A spatial-data workflow can therefore be defined as an organized sequence of computational and human activities through which geographic information is collected, ingested, validated, transformed, filtered, classified, analyzed, visualized, quality controlled, stored, and distributed. Within enterprise environments, spatial-data workflows frequently operate across multiple software applications, computing resources, databases, storage environments, and user roles. Enterprise LiDAR platforms represent integrated technological environments designed to support scalable processing, management, analysis, governance, and distribution of large LiDAR datasets across organizational operations (Franciosa et al., 2020). Workflow redesign refers to the systematic restructuring of tasks, decision points, computational processes, information exchanges, and human-system interactions to improve measurable operational performance. Machine learning enhanced workflow redesign extends this concept by incorporating computational models capable of learning patterns from historical or labeled data and applying those patterns to classification, segmentation, anomaly detection, parameter selection, quality assessment, prioritization, and other spatial-processing decisions. Task success represents the degree to which a predefined spatial-processing objective is completed correctly according to established technical and operational criteria. Operational cost represents the measurable resources consumed in completing the workflow, including employee time, computation, storage, network utilization, quality assurance, error correction, and repeated processing (Hou & Ai, 2020). A quantitative evaluation of machine learning enhanced spatial-data workflow redesign consequently involves systematic numerical measurement of differences between conventional and redesigned workflow conditions. Relevant indicators can include successful task completion rates, execution time, processing failures, manual interventions, computational utilization, rework frequency, and monetary cost per completed task. These definitions establish a unified framework in which machine learning is evaluated not only as an analytical technology but also as a component of an enterprise operational process whose performance can be observed through measurable effectiveness and resource-efficiency outcomes (Gallwey et al., 2019).

LiDAR technology has acquired substantial international significance because governments, commercial organizations, research institutions, infrastructure operators, environmental agencies, and technology companies increasingly depend on high-resolution three-dimensional spatial information for planning and operational decision-making. Large-scale LiDAR acquisition programs support national elevation mapping, flood modeling, transportation planning, forest assessment, coastal monitoring, geological investigation, agricultural management, urban development, energy infrastructure, construction, disaster management, and environmental protection (Long et al., 2018). Airborne, terrestrial, mobile, drone-based, and spaceborne LiDAR systems have expanded the geographic scale and diversity of available three-dimensional observations. National mapping initiatives generate extensive elevation datasets that support public administration and infrastructure development, while commercial organizations increasingly use LiDAR for asset inspection, engineering analysis, autonomous systems, digital twins, logistics, telecommunications, and location-based intelligence. The international expansion of smart-city initiatives has further increased demand for accurate three-dimensional representations of roads, buildings, utilities, vegetation, and public infrastructure (Li et al., 2020). Transportation authorities use point-cloud information to evaluate road corridors, bridges, railways, clearances, and roadside assets. Forestry organizations employ LiDAR measurements to estimate canopy structure, vegetation height, biomass-related characteristics, and forest composition. Disaster-management organizations use high-resolution terrain information for

flood assessment, landslide analysis, coastal vulnerability evaluation, and post-event damage mapping. Construction and engineering organizations apply LiDAR to site documentation, progress monitoring, structural assessment, and comparison between physical environments and digital models. These applications generate substantial volumes of spatial information that must be converted from raw observations into operationally meaningful products (Okaro et al., 2019).

**Figure 1: Machine Learning Enhanced LiDAR Workflow**



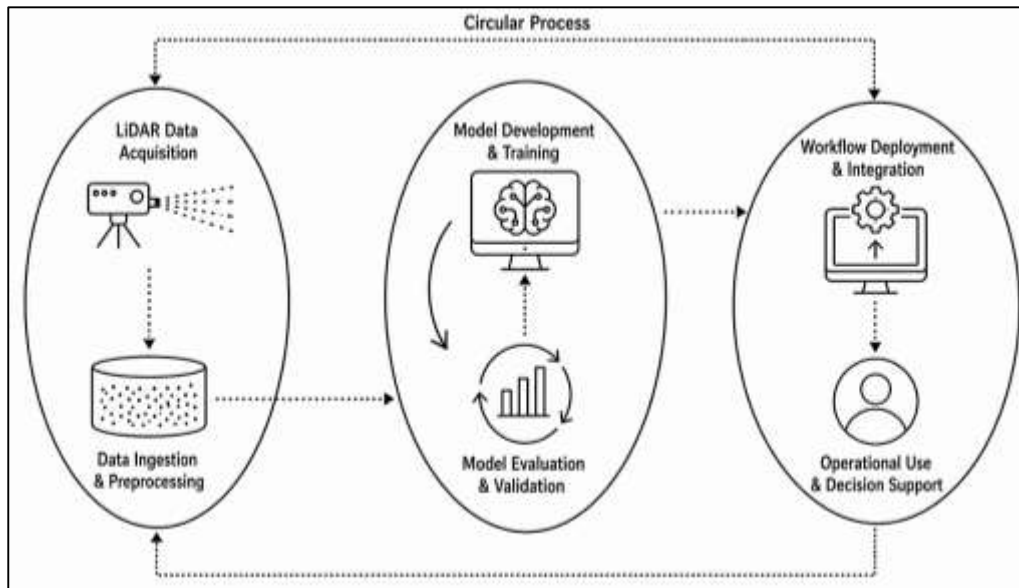
As LiDAR coverage expands geographically and point densities increase, organizations encounter greater demands for scalable storage, computational capacity, rapid processing, standardized quality control, and repeatable analytical workflows. International enterprises operating across multiple regions face additional requirements involving heterogeneous sensors, varying coordinate-reference systems, different acquisition specifications, inconsistent data structures, distributed processing teams, and geographically dispersed computing resources. The value of LiDAR therefore depends not only on the accuracy of data acquisition but also on the organizational capacity to transform large point-cloud datasets into usable spatial information efficiently. Machine learning has become increasingly relevant within this environment because many traditional LiDAR operations depend on repetitive classification, filtering, object identification, feature extraction, and quality-review activities (Li et al., 2015). Automating selected activities through data-driven models can restructure the relationship between human analysts and computational systems. International significance is consequently associated with the growing operational dependence on three-dimensional spatial information and the need to process increasingly large and complex LiDAR datasets while maintaining acceptable levels of accuracy, consistency, speed, and cost.

Enterprise LiDAR processing involves considerably more than executing a single classification or visualization algorithm. A complete spatial-data workflow can include data acquisition, file transfer, ingestion, metadata verification, coordinate-reference validation, trajectory adjustment, registration, noise removal, spatial indexing, tiling, ground filtering, height normalization, geometric feature generation, segmentation, classification, object extraction, surface modeling, accuracy assessment, quality assurance, visualization, storage, and final product distribution (Yang et al., 2020). Each stage can involve different software tools, data structures, computational resources, technical specialists, and quality requirements. Large point-cloud datasets create additional complexity because individual projects may contain millions, billions, or substantially larger numbers of three-dimensional observations. Processing such datasets can require substantial memory, computing capacity, storage space, and input-output operations. Organizations frequently divide large datasets into tiles or geographic subsets to make processing computationally manageable. These subdivisions can create additional workflow activities related to indexing, naming conventions, spatial boundaries, data duplication, merging, and consistency verification. Enterprise platforms must also accommodate differences among airborne laser scanning, terrestrial laser scanning, mobile mapping systems, drone-based acquisition, and other sensor configurations (Siahpour, et al., 2020). Variations in point density, scanning geometry, return characteristics, noise, spatial coverage, and environmental conditions can affect the reliability of standardized processing procedures. Manual intervention remains common

when users must select parameters, inspect intermediate outputs, correct classifications, identify processing failures, or transfer data between disconnected applications. Repeated handoffs between systems can increase execution time and create opportunities for configuration errors, duplicated processing, missing files, inconsistent outputs, and unnecessary data movement. Workflow fragmentation can also occur when specialized algorithms operate independently from enterprise databases, storage systems, user interfaces, and quality-control procedures. Under such conditions, improvements in one computational stage may produce limited reductions in overall task duration because bottlenecks remain elsewhere in the process. Workflow redesign addresses this issue by examining the complete sequence of activities rather than optimizing individual algorithms in isolation (Jia, et al., 2020). A redesigned workflow may alter the ordering of tasks, automate repetitive operations, reduce unnecessary transfers, standardize parameter selection, consolidate validation procedures, introduce exception-based review, and connect analytical outputs directly with subsequent processing stages. Machine learning provides an additional mechanism for redesign by replacing or supporting selected rule-based and manually intensive decisions. The complexity of enterprise LiDAR workflows therefore creates a quantitative management problem in which task success and operational cost are influenced simultaneously by algorithm performance, data characteristics, system architecture, human intervention, processing sequence, and computational resource consumption (Soroush et al., 2020).

Machine learning has transformed the processing of three-dimensional spatial information by enabling computational systems to identify complex patterns within point-cloud data and perform analytical activities that previously required extensive handcrafted rules or manual interpretation. Traditional LiDAR processing commonly relies on geometric thresholds, morphological operations, manually selected features, predetermined neighborhood structures, and domain-specific processing parameters. Machine learning expands these capabilities by learning relationships directly from labeled or previously processed observations. Algorithms can be trained to distinguish terrain from nonground objects, identify buildings and vegetation, recognize transportation infrastructure, segment urban objects, detect anomalies, estimate object characteristics, and evaluate the quality of processed spatial information (Aubreville et al., 2017). Deep learning further extends this capability by learning hierarchical representations from raw or minimally processed point-cloud observations. Modern three-dimensional learning architectures can operate directly on unordered points, structured spatial neighborhoods, voxel representations, projected range images, sparse tensors, or combinations of multiple representations. These approaches allow complex spatial structures to be captured without requiring every relevant characteristic to be manually designed in advance. Within enterprise platforms, the importance of machine learning extends beyond predictive accuracy because learned models can be positioned at multiple stages of the operational workflow. A model can automatically recommend processing parameters, identify potentially problematic data, classify point-cloud observations, prioritize uncertain outputs for human inspection, detect failed processing stages, or route datasets through different processing sequences according to their characteristics (Rishav et al., 2020). Confidence scores can be used to distinguish routine observations from uncertain cases, allowing analysts to concentrate on outputs that require professional judgment rather than reviewing every processed observation equally. This changes the role of automation from simple replacement of individual manual activities toward selective allocation of human attention. Machine learning can also support consistency because similar input patterns can be processed using standardized learned decision functions rather than relying entirely on differences in individual operator choices. Quantitative evaluation remains essential because higher model accuracy does not necessarily translate directly into higher workflow task success. A highly accurate model can still produce inefficient operational outcomes when preprocessing is excessive, computational requirements are high, integration is difficult, or outputs require substantial manual correction (Shi et al., 2019). The appropriate unit of analysis is therefore the end-to-end task in which model execution represents one component of a larger operational sequence. Machine learning enhanced workflow redesign can be evaluated by determining whether integration of automated analytical functions increases the probability of successful task completion while reducing processing duration, human intervention, repeated execution, resource utilization, and associated operational expenditures.

**Figure 2: Machine Learning LiDAR Workflow Framework**



Operational cost represents a critical performance dimension in enterprise LiDAR platforms because three-dimensional spatial processing consumes multiple categories of organizational resources. Human labor constitutes an important component because analysts, technicians, engineers, data scientists, and quality-control personnel may spend substantial time preparing datasets, configuring algorithms, monitoring processing, validating results, correcting classifications, resolving errors, and generating final products (Meyes et al., 2017). Computational resources create another cost category through processor utilization, graphics-processing capacity, memory consumption, cloud-computing charges, and execution time. Storage expenses can become substantial when organizations maintain raw LiDAR observations, intermediate files, tiled datasets, derived surfaces, classification outputs, backups, and multiple versions of processed products. Network transfer and data movement can also generate direct and indirect costs, particularly when very large point-cloud datasets are transferred repeatedly between local systems, cloud environments, distributed teams, and specialized processing applications. Workflow failures introduce additional economic burdens because unsuccessful execution can require troubleshooting, parameter modification, reprocessing, additional quality control, and repeated data transfers. Manual correction creates further cost when automated outputs fail to satisfy established acceptance criteria. These components demonstrate why operational cost should be evaluated across the complete workflow rather than measured exclusively through the price of computational infrastructure or software (Kim et al., 2020). Machine learning enhanced workflow redesign can affect each cost category differently. Automated classification may reduce analyst time while increasing computational requirements. Improved error detection may add a processing stage while reducing expensive downstream rework. Automated parameter selection can reduce configuration time and improve standardization, while confidence-based review can decrease the volume of outputs requiring manual inspection. Spatial indexing and intelligent workload selection can reduce unnecessary computation by limiting processing to relevant geographic areas or data subsets. Workflow orchestration can minimize redundant file movement and automatically connect successful outputs with subsequent processing activities. The economic value of these changes depends on their combined effects on the cost required to produce an accepted output. Cost per successfully completed task therefore provides a useful quantitative measure because it integrates resource consumption with operational effectiveness (Pfeiffer et al., 2018). Other relevant measures include analyst minutes per task, computational hours, storage consumption, number of processing attempts, error-correction time, manual review duration, and rework frequency. Comparing these indicators between conventional and redesigned workflows allows operational efficiency to be measured numerically and prevents conclusions based solely on technological sophistication. In this framework,

cost reduction represents a measurable change in resources consumed during execution rather than an assumed benefit associated with automation.

Task success provides a central effectiveness measure for evaluating whether an enterprise spatial-data workflow performs its intended function under predefined operational conditions (Wahab et al., 2020). In LiDAR environments, successful task completion can require more than technically executing an algorithm. A task may be considered successful only when the required input data are correctly identified, preprocessing is completed without critical error, the analytical operation executes correctly, output accuracy meets established thresholds, required products are generated, and quality-control requirements are satisfied. Task success can therefore be represented as a binary variable indicating successful or unsuccessful completion, as a proportion of successfully completed tasks, or through a composite performance index incorporating several acceptance criteria. This outcome can be supplemented by continuous measures including total completion time, active human-processing time, automated-processing time, number of user interactions, error frequency, number of retries, manual corrections, computational resource consumption, and cost (Amorin et al., 2019). Such measures allow machine learning enhanced workflow redesign to be assessed from both effectiveness and efficiency perspectives. Quantitative measurement is especially important because LiDAR tasks vary substantially in complexity. Dataset size, geographic extent, point density, sensor type, number of target classes, environmental characteristics, object complexity, and required output quality can influence task performance independently of workflow design. These characteristics can therefore be measured as control variables when comparing workflow conditions. User-related characteristics can also influence outcomes when analysts possess different levels of experience with LiDAR processing, enterprise software, or machine-learning tools. Standardized task instructions and comparable datasets can reduce uncontrolled variation during evaluation. Platform-generated logs provide another valuable source of quantitative evidence because execution timestamps, system errors, resource consumption, processing status, and user interactions can be recorded automatically. These objective records can complement manually recorded task outcomes and cost information (Shevchik et al., 2020). A baseline workflow provides the reference condition representing existing operational practice, while the redesigned condition incorporates machine-learning-supported activities and modified process structures. Differences between these conditions can then be evaluated through descriptive statistics and inferential statistical procedures. Task-success proportions can be compared across workflow conditions, while processing time, operational cost, manual intervention, and computational consumption can be evaluated as continuous outcomes. Multivariate analysis can further account for workload characteristics and other measured factors affecting performance (Xiang et al., 2019). This quantitative structure separates technological claims from empirically observed operational results by requiring workflow performance to be represented through numerical evidence obtained under defined evaluation conditions.

The research problem addressed by Machine Learning Enhanced Spatial-Data Workflow Redesign: A Quantitative Evaluation of Task Success and Operational Cost Reduction in Enterprise LiDAR Platforms emerges from the growing difference between the technical capabilities of machine-learning algorithms and the need for measurable evidence concerning their performance within complete enterprise spatial-data workflows. Machine-learning models can achieve high levels of point-cloud classification, segmentation, feature identification, and object recognition, yet enterprise performance depends on a broader combination of data preparation, computational execution, user interaction, quality assurance, exception management, storage, and product delivery (Braun & Borrmann, 2019). Organizations therefore require quantitative evidence that distinguishes algorithmic performance from workflow-level operational performance. A machine-learning component may improve one processing stage while leaving substantial manual activities elsewhere in the workflow, or it may reduce human processing requirements while increasing computational resource consumption. Evaluation at the complete task level provides a means of capturing these interactions. The present quantitative framework centers on two primary outcomes: task success and operational cost. Task success represents the effectiveness dimension by measuring whether assigned LiDAR processing activities are completed according to predetermined acceptance requirements. Operational cost represents the resource-efficiency dimension by measuring labor, computation, storage, reprocessing, quality-control

activity, and other quantifiable resources consumed during task execution (Paul et al., 2019). The independent workflow condition can be structured around conventional and machine learning enhanced processing configurations, allowing measurable differences between conditions to be identified. Additional performance variables can include elapsed processing time, analyst involvement, number of manual actions, failure frequency, number of processing attempts, correction requirements, and computational utilization. Dataset characteristics can be recorded to account for variation in processing complexity, including point-cloud size, density, spatial extent, classification requirements, and task category. Statistical analysis can then determine the magnitude and significance of differences in task-success rates and operational costs associated with workflow redesign while controlling for relevant operational factors. This structure places the enterprise workflow rather than the individual machine-learning model at the center of quantitative evaluation (Camburn et al., 2020). The resulting research design treats machine learning as an embedded operational mechanism whose contribution is measured through observable changes in completed tasks and consumed resources, establishing a clearly defined empirical relationship among workflow configuration, task performance, and enterprise LiDAR processing cost (Chun-Lei et al., 2019).

The primary objective of this quantitative study is to evaluate the effectiveness of machine learning enhanced spatial-data workflow redesign in improving task success and reducing operational costs within enterprise LiDAR platforms. The study specifically seeks to determine whether the integration of machine learning into established LiDAR processing workflows produces measurable improvements in the successful completion of spatial-data tasks when compared with conventional workflow configurations. Task success is examined through quantifiable indicators such as completion accuracy, successful processing rate, error frequency, processing failures, number of repeated attempts, manual correction requirements, and compliance with predefined output-quality criteria. A further objective is to measure the extent to which the redesigned workflow affects operational efficiency by examining total processing time, analyst involvement, frequency of manual interactions, computational resource utilization, data-processing repetitions, quality-control effort, and workflow rework. The study also aims to quantify operational cost reduction associated with machine learning enhanced processing by evaluating measurable resource expenditures across the complete LiDAR workflow rather than focusing exclusively on the computational cost of individual algorithms. Accordingly, labor expenditure, computing consumption, storage requirements, processing duration, error-correction activities, and repeated execution are considered important components of operational cost. Another objective is to compare baseline and machine learning enhanced workflow conditions under comparable enterprise processing tasks in order to identify statistically significant differences in effectiveness and cost-related outcomes. The study further seeks to examine whether characteristics of LiDAR workloads, including point-cloud volume, point density, spatial extent, processing complexity, and task category, influence the relationship between workflow redesign and operational performance. In addition, the research aims to determine whether reductions in manual intervention are associated with improvements in task completion efficiency and whether machine-learning-supported automation can maintain required quality standards while decreasing resource consumption. The quantitative evaluation also seeks to establish the magnitude of performance differences between workflow configurations through appropriate descriptive and inferential statistical measures, thereby allowing improvements to be expressed in numerical terms such as percentage changes in task success, processing duration, intervention frequency, and cost per successfully completed task. Collectively, these objectives establish a measurable framework for assessing machine learning as an integrated component of enterprise spatial-data operations and for determining how workflow configuration is associated with task-level effectiveness, resource utilization, and operational cost within large-scale LiDAR processing environments.

## **LITERATURE REVIEW**

The literature surrounding machine learning enhanced spatial-data workflow redesign occupies an interdisciplinary research domain connecting LiDAR technology, geographic information systems, three-dimensional point-cloud processing, machine learning, enterprise information systems, workflow automation, computational efficiency, human-computer interaction, and operational cost management. The rapid expansion of LiDAR acquisition has resulted in increasingly large, dense,

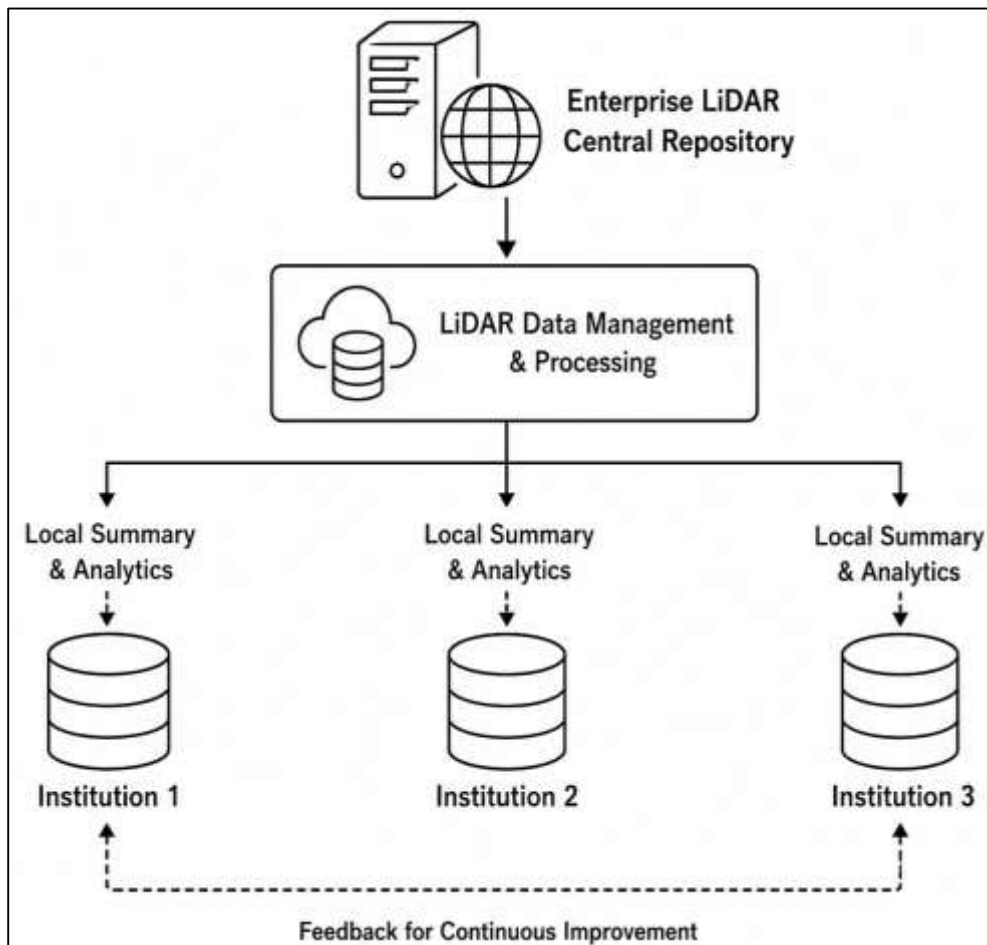
heterogeneous, and computationally demanding spatial datasets, creating significant processing requirements for organizations that depend on three-dimensional geographic information (Debasish, 2021; Jing et al., 2018). Enterprise LiDAR platforms commonly support sequential activities involving data ingestion, preprocessing, filtering, spatial indexing, segmentation, classification, feature extraction, quality assessment, visualization, storage, and output generation. The performance of these platforms is consequently influenced not only by the predictive accuracy of individual algorithms but also by processing latency, computational utilization, manual intervention, workflow failures, rework, data-transfer requirements, and the cost required to generate an accepted spatial product. Machine learning has become particularly important within this environment because automated models can perform or support activities that traditionally depend on rule-based processing and repeated human decision-making. Existing technical research demonstrates considerable progress in point-cloud classification and semantic segmentation, while computational studies increasingly evaluate the balance among predictive accuracy, model complexity, processing speed, memory utilization, and real-time capability (Abdur & Iftekhhar, 2021; Sampredo et al., 2018). For example, contemporary LiDAR segmentation research explicitly treats accuracy, parameter requirements, and running time as interconnected performance dimensions, reinforcing the importance of evaluating efficiency alongside predictive performance.

For the present quantitative study, the literature requires a broader analytical structure than a conventional review centered exclusively on machine-learning model accuracy. Research on large-scale point-cloud processing indicates that the irregular, sparse, and unstructured characteristics of LiDAR data create substantial computational challenges, particularly when processing extensive outdoor or enterprise datasets. Research also demonstrates that alternative representations and lightweight processing architectures can substantially change inference time and computational requirements; one reported lightweight approach, for instance, reduced model parameters by 87% and demonstrated measurable improvements in processing and power efficiency (Isele et al., 2018; Sheak & Risha, 2022). More recent approaches similarly report simultaneous improvements in segmentation accuracy and inference speed, demonstrating that computational architecture can influence both technical effectiveness and resource efficiency. These findings establish an important basis for examining enterprise workflow redesign, where the relevant dependent variables extend from algorithmic accuracy to task success, processing duration, manual intervention, computational resource consumption, rework, and operational cost. Accordingly, this literature review is structured to move systematically from the characteristics and computational demands of enterprise LiDAR data toward machine-learning automation, workflow redesign, task-success measurement, operational efficiency, cost reduction, and quantitative evaluation (Chowdhury & Tonay, 2022; Zhu & Pierskalla Jr, 2016). Particular emphasis is placed on identifying measurable relationships among workflow configuration, automation intensity, task completion, processing time, error occurrence, human effort, resource utilization, and cost per successfully completed task. This structure establishes a literature foundation aligned directly with the variables and empirical orientation of the study rather than treating machine learning, LiDAR processing, and enterprise workflow management as independent research areas (Niroui et al., 2019; Nishat & Kaniz, 2022).

### **Enterprise LiDAR Spatial-Data Processing**

Enterprise LiDAR spatial-data processing is fundamentally shaped by the quantitative characteristics of point-cloud datasets, particularly point count, point density, spatial extent, dimensionality, file size, sparsity, irregularity, acquisition frequency, and attribute complexity. The literature consistently characterizes LiDAR point clouds as computationally distinctive because they consist of large collections of three-dimensional observations distributed irregularly throughout geographic space rather than arranged within the fixed structures associated with conventional raster imagery (Hegeman et al., 2014; Shaiful et al., 2022). A single LiDAR observation commonly contains three-dimensional coordinates together with additional attributes such as intensity, return number, classification, scan angle, timestamp, color information, source identifier, and sensor-specific measurements. Consequently, growth in point count increases not only the number of spatial coordinates that must be processed but also the total volume of associated attribute information that must be stored, retrieved, transformed, and analyzed.

**Figure 3: Enterprise LiDAR Data Processing Framework**



Multiple studies of large-scale point-cloud classification and semantic segmentation have demonstrated that dataset scale is a major determinant of computational difficulty. Established benchmark datasets used in three-dimensional spatial research contain millions to billions of points distributed across urban streets, buildings, vegetation, transportation infrastructure, terrain, and other object categories. Enterprise datasets can become substantially larger when repeated surveys, extensive geographic coverage, high acquisition frequencies, or multiple sensors are incorporated into a common spatial-data environment (Li, 2020). Point density creates another important quantitative dimension because higher densities increase the number of observations representing a given geographic area and therefore expand memory, storage, and processing requirements. Spatial extent interacts with density because high-density acquisition over extensive geographic regions produces substantially larger point counts than either characteristic independently. The literature on point-based neural networks, large-scale semantic segmentation, sparse convolution, random-sampling architectures, and spatial partitioning repeatedly identifies large point populations as a fundamental computational challenge. Irregular point distribution further complicates processing because conventional grid-based computational structures cannot always be applied directly without transformation, voxelization, projection, sampling, or neighborhood construction. Sparsity varies across geographic space and sensor configurations, requiring processing systems to manage uneven concentrations of observations (Lisánczuk et al., 2020). Enterprise LiDAR workload should therefore be understood as a multidimensional quantitative phenomenon in which point count, density, coverage, dimensionality, attribute volume, and spatial irregularity collectively determine the scale of data that computing and storage infrastructure must manage.

The quantitative structure of LiDAR data varies considerably across airborne, terrestrial, mobile, drone-based, and vehicle-mounted acquisition systems, making acquisition configuration an important determinant of enterprise processing workload. Airborne LiDAR is commonly designed for relatively extensive geographic coverage and is frequently employed for terrain mapping, forestry, flood assessment, infrastructure planning, coastal analysis, and regional elevation modeling (Li et al., 2018). Its enterprise processing burden is often generated through the combination of broad spatial extent and substantial cumulative point counts. Terrestrial laser scanning operates from stationary ground positions and can generate exceptionally dense measurements of buildings, industrial facilities, construction sites, geological structures, vegetation, and other localized environments. Although terrestrial surveys may cover smaller geographic areas than airborne campaigns, their very high local density can produce large datasets that require substantial memory and processing capacity. Drone-based LiDAR occupies an intermediate operational position because low-altitude acquisition can produce dense point clouds across project-scale areas while supporting repeated data collection. Repeated drone surveys used for construction monitoring, asset inspection, topographic assessment, and environmental observation can substantially increase enterprise storage requirements because organizations may retain multiple temporal versions of the same geographic environment (Wang et al., 2017). Mobile and vehicle-mounted LiDAR systems generate another distinctive quantitative workload because observations are acquired continuously along transportation corridors and urban environments. Such systems can produce dense sequential measurements containing geometric coordinates, intensity values, timestamps, trajectory-related information, and additional sensor attributes. Research using large urban and road-scene datasets has demonstrated that mobile and vehicle-based acquisition creates extensive point populations containing buildings, roads, vehicles, vegetation, poles, signs, pedestrians, and other geometrically diverse objects. The resulting class complexity creates additional computational requirements for segmentation and classification. Studies of urban point-cloud benchmarks also demonstrate substantial variations in object frequency and class distribution, meaning that dataset size alone does not fully represent processing difficulty (Li et al., 2019). A billion-point dataset containing relatively homogeneous terrain presents a different computational and analytical workload from a similarly sized dataset containing numerous small and geometrically complex urban objects. Acquisition frequency further changes enterprise workload because repeated mapping generates temporal point-cloud collections that must be stored, indexed, compared, and quality controlled. The literature therefore establishes acquisition platform, geographic coverage, density, temporal frequency, sensor configuration, and environmental complexity as interconnected quantitative factors that determine the volume and difficulty of enterprise LiDAR processing (Vo et al., 2019).

Increasing point-cloud volume and complexity directly influence computational workload through processing time, memory demand, processor utilization, graphics-processing requirements, data-transfer operations, and achievable throughput. Numerous studies examining large-scale point-cloud processing have identified memory management as a central constraint because complete datasets cannot always be loaded into available system memory or graphics memory simultaneously. This constraint has contributed to the widespread use of spatial tiling, sampling, hierarchical partitioning, sparse representations, voxel structures, neighborhood selection, and block-based processing. Point-based machine-learning methods must frequently construct local neighborhoods or determine relationships among nearby points, creating computational operations whose cost increases as the number of observations grows (Hegeman et al., 2014). Large-scale semantic segmentation research has consequently placed considerable emphasis on efficient sampling and local feature aggregation because exhaustive processing of extremely large point populations can produce substantial execution times and memory requirements. Sparse-convolution approaches address a related problem by concentrating computation on occupied spatial locations rather than processing every position within a dense three-dimensional grid. Other approaches transform three-dimensional observations into two-dimensional representations to obtain computational efficiencies associated with established image-processing structures, although such transformations can alter spatial information and introduce additional preprocessing requirements.

The literature comparing point-based, voxel-based, projection-based, sparse, and hybrid architectures therefore demonstrates that representation strategy materially influences processing throughput and resource utilization (Richter & Döllner, 2014). Processing time per million points provides a particularly useful enterprise measure because it standardizes computational performance across datasets of different sizes. Points processed per second similarly provides a throughput indicator that can be compared across algorithms, computing configurations, and workflow stages. CPU and GPU utilization are also important because machine-learning-enhanced LiDAR workflows may shift computational demand from conventional processors toward graphics-processing resources. Memory utilization becomes particularly significant during training and inference when neighborhood features, intermediate representations, model parameters, and prediction outputs must coexist with point-cloud data. Input-output operations represent an additional workload component because large datasets must repeatedly be read from storage and written as filtered, classified, tiled, indexed, or transformed outputs (Li et al., 2020). Several studies of scalable spatial computing and point-cloud processing indicate that data movement itself can become a performance bottleneck when computational operations are distributed across storage and processing environments. Computational workload in enterprise LiDAR systems should therefore be measured through multiple quantitative indicators rather than processing time alone, including memory consumption, CPU/GPU utilization, throughput, input-output activity, and processing time normalized by point count (Li et al., 2016).

Storage consumption and data movement represent major components of enterprise LiDAR workload because increases in point count, attribute complexity, acquisition frequency, and geographic coverage produce corresponding growth in raw and derived data volumes. Enterprise LiDAR repositories rarely contain only the original point clouds. Operational workflows commonly generate multiple representations and intermediate products, including cleaned datasets, spatial tiles, classified point clouds, normalized elevations, digital terrain models, digital surface models, extracted objects, machine-learning predictions, quality-control files, visualization products, metadata, backups, and archived versions. The effective storage requirement of a LiDAR project can therefore exceed the size of its initial acquisition files (Jung & Pijanowski, 2015). File format and compression influence this requirement because compressed representations can reduce persistent storage and transfer volume, while decompression and conversion introduce additional processing activity. Literature addressing spatial indexing, hierarchical data organization, point-cloud databases, cloud-based processing, and distributed spatial computing consistently demonstrates the importance of minimizing unnecessary movement of large spatial datasets. When datasets reach hundreds of gigabytes or terabyte scale, repeated transfer between local storage, network systems, cloud infrastructure, processing nodes, and analyst workstations can materially increase input-output latency and operational workload. Spatial partitioning and indexing reduce some of these demands by allowing systems to retrieve geographic subsets rather than repeatedly accessing complete collections. Tiling similarly enables large datasets to be divided into computationally manageable units, although excessive fragmentation can increase file-management complexity and generate additional merging or boundary-processing activities (Nguyen et al., 2020). Enterprise scalability is therefore influenced by the relationship among dataset size, storage architecture, network capacity, computing resources, processing strategy, and workflow design. Research on distributed geospatial processing has shown that partitioning and parallel execution can improve the management of large spatial workloads, while point-cloud studies have demonstrated the importance of efficient sampling, sparse computation, and scalable neighborhood processing for extensive three-dimensional datasets. More than ten established research streams covering large-scale semantic segmentation, point-based learning, sparse convolution, urban point-cloud benchmarking, airborne laser scanning, mobile mapping, spatial big-data processing, point-cloud libraries, scalable geospatial computing, and LiDAR data management collectively support this interpretation (Liang et al., 2015). Quantitative assessment of enterprise scalability should consequently include point count, points per square meter, total dataset size in gigabytes or terabytes, spatial coverage, memory consumption, CPU and GPU utilization, processing time per million points, points processed per second, input-output volume, and total storage consumption. These measures provide a comprehensive empirical representation of how the quantitative growth of LiDAR data translates into enterprise computational workload (Awrangjeb, 2015).

### **Conventional Spatial-Data Workflows in Enterprise LiDAR Processing**

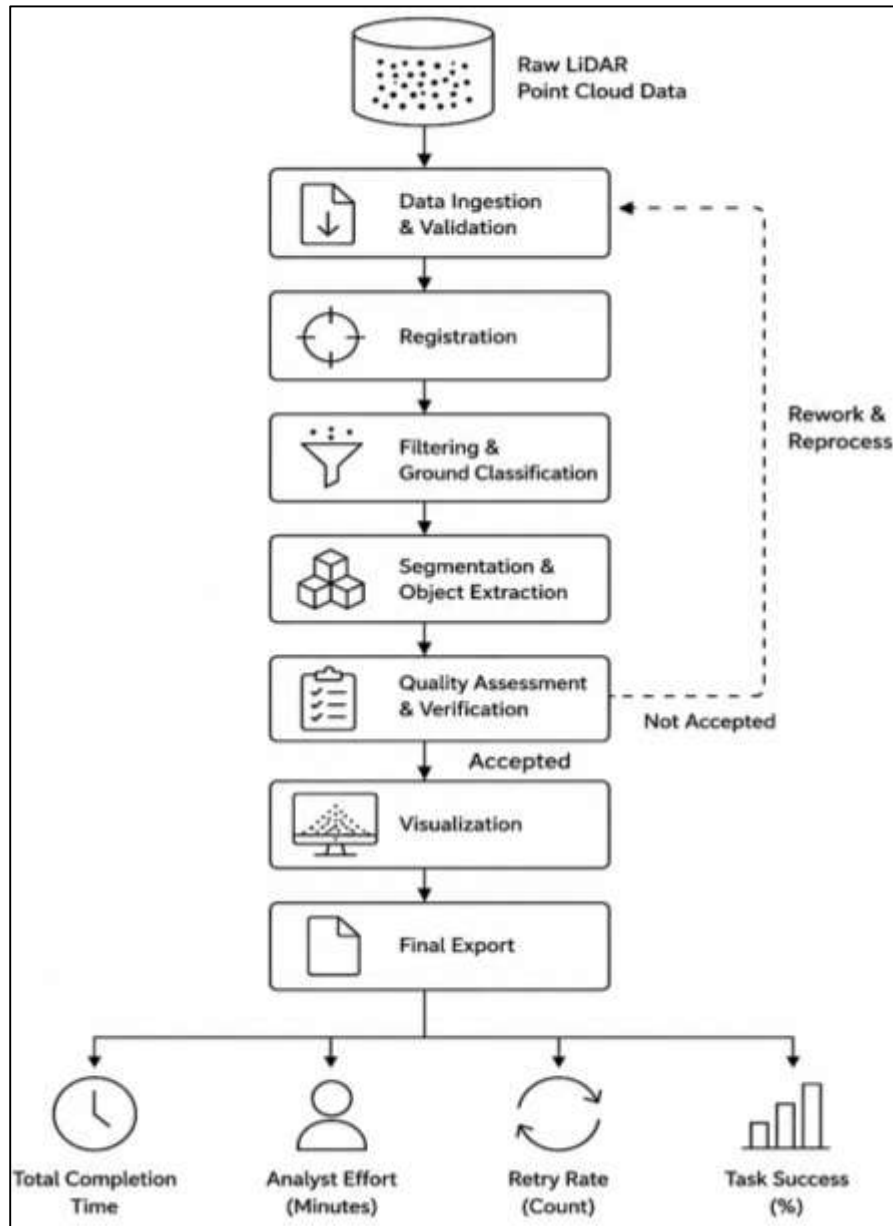
Conventional enterprise LiDAR workflows provide an important baseline for evaluating whether machine learning enhanced workflow redesign produces measurable improvements in operational performance. The literature characterizes conventional LiDAR processing as a multistage sequence in which raw point-cloud data pass through ingestion, data validation, coordinate and trajectory verification, registration, noise removal, filtering, ground classification, segmentation, feature or object extraction, quality assessment, visualization, and final export. More than ten studies examining airborne laser scanning, terrestrial laser scanning, mobile mapping, point-cloud classification, ground filtering, segmentation, registration, and geospatial data management collectively demonstrate that these stages are technically interdependent because the quality and efficiency of an earlier operation can influence the processing requirements of subsequent activities (Lei et al., 2019). Data ingestion represents an initial workload because large files must be transferred, organized, indexed, checked for completeness, and associated with appropriate metadata and coordinate-reference information. Preprocessing can involve removal of erroneous observations, coordinate transformation, point-density assessment, trajectory correction, and preparation of tiles or processing blocks. Registration introduces additional computational and human effort when multiple scans or acquisition strips must be aligned before further analysis. Filtering and ground classification then separate relevant surfaces and object categories, while segmentation and object extraction transform point-level observations into analytically meaningful entities. Conventional workflows often rely on different tools for these activities, producing software handoffs and intermediate outputs that increase the number of processing stages (Yue et al., 2020). Quantitative performance is therefore determined by the complete elapsed time from initial data ingestion to accepted output rather than by the execution time of one algorithm. Processing time per task can increase substantially when analysts must wait for large files to load, manually configure processing parameters, inspect intermediate results, repeat unsuccessful stages, and convert outputs into formats required by subsequent applications. The number of workflow steps is consequently an important baseline indicator because longer processing chains generally create more opportunities for delay, interaction, and failure. Literature on large-scale point-cloud processing also demonstrates that dataset size, point density, spatial extent, computing resources, and algorithmic complexity influence execution duration (Yang et al., 2020). Baseline workflow performance can therefore be represented quantitatively through total completion time, processing time per stage, number of workflow steps, task-success percentage, and resource consumption required to produce an accepted enterprise LiDAR output.

Human effort represents a substantial component of conventional LiDAR workflow performance because many processing activities require analysts to configure parameters, select data, review outputs, correct classifications, manage files, resolve errors, and determine whether intermediate products satisfy established quality requirements. The literature on point-cloud processing consistently identifies manual and semi-automated activities across registration, filtering, classification, segmentation, feature extraction, and quality assurance (Li et al., 2018). Ground-filtering procedures, for example, can require parameters related to slope, elevation differences, neighborhood size, window dimensions, or surface characteristics, while segmentation procedures can depend on thresholds governing distance, normal vectors, geometric similarity, or minimum object size. Registration may require selection of reference scans, control information, initial alignment, correspondence verification, and inspection of residual errors. These activities create measurable analyst minutes per task even when the principal computational operation itself is automated. Human effort increases when the same parameters cannot be applied reliably across heterogeneous datasets containing different terrain conditions, point densities, sensor characteristics, urban structures, vegetation patterns, or acquisition geometries. Software handoffs add another source of operational workload (Hegeman et al., 2014).

Enterprise LiDAR processing frequently combines specialized point-cloud applications, GIS environments, database systems, visualization tools, scripting utilities, and quality-control software. Moving information among these environments can require file conversion, export, import, coordinate verification, directory management, attribute checking, and manual initiation of subsequent stages. Such handoffs may not improve the spatial product directly, yet they contribute to overall completion time and operational cost. Repetitive user interactions also accumulate across large projects when

analysts perform identical setup procedures for numerous tiles, scan regions, flight strips, or project areas (Xie et al., 2019).

**Figure 4: Conventional Enterprise LiDAR Workflow Performance**



Studies addressing workflow automation, point-cloud software environments, spatial data infrastructures, and large-scale geospatial processing have emphasized the operational importance of reducing unnecessary data movement and repeated processing activities (Tang & Feng, 2017). From a quantitative perspective, analyst effort can be measured through active handling minutes, number of manual interactions, number of configuration decisions, software transitions, and quality-control interventions per task. These indicators provide a more complete baseline than algorithmic accuracy alone because a conventional workflow can produce technically acceptable outputs while consuming substantial employee time. Baseline operational cost per completed task is therefore closely associated with both computational processing and the accumulated human effort required to coordinate individual processing stages (Wang et al., 2019).

Workflow reliability represents another major quantitative dimension of conventional enterprise LiDAR processing because errors and unsuccessful intermediate outputs can generate repeated

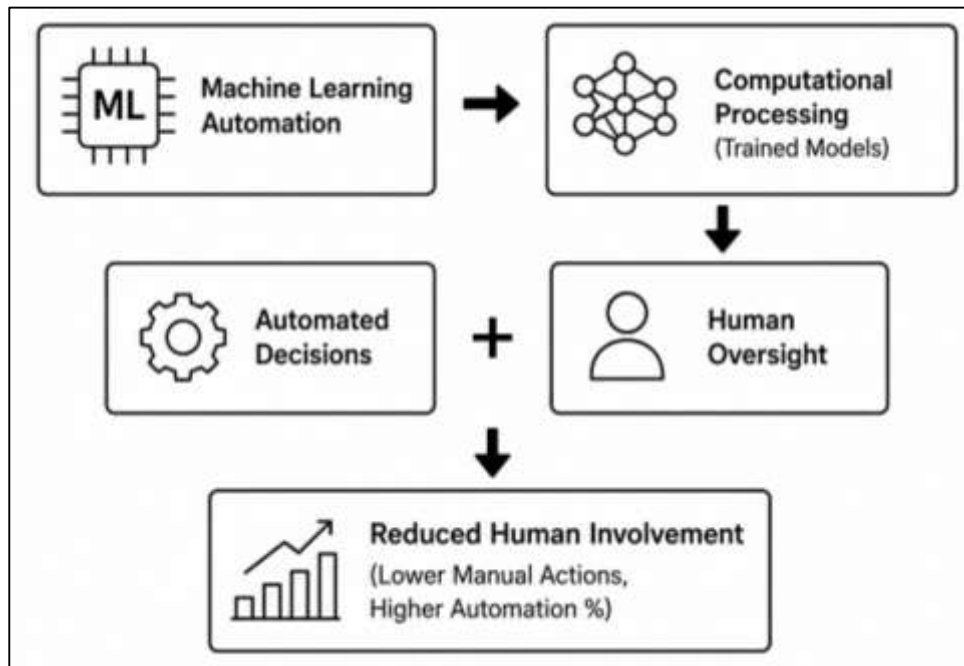
processing, additional analyst effort, and higher resource consumption. Research across LiDAR registration, filtering, point-cloud classification, segmentation, feature extraction, and quality assessment demonstrates that processing outcomes are sensitive to data quality, parameter selection, point density, environmental complexity, sensor characteristics, noise, occlusion, and class ambiguity (Omidipour et al., 2020). An error introduced during an early processing stage can propagate through later activities and may not become visible until classification, object extraction, visualization, or final quality control. Incorrect registration can influence geometric consistency, unsuitable filtering parameters can remove valid observations or retain unwanted points, and inaccurate ground classification can affect elevation-derived products and subsequent object segmentation. Conventional quality-control procedures consequently require analysts to inspect intermediate or final outputs and determine whether established accuracy requirements have been satisfied. When results fail these requirements, analysts may return to an earlier processing stage, modify parameters, rerun the operation, regenerate downstream products, and conduct another quality assessment. This creates a rework cycle that can materially increase completion time and cost (Li, 2020). Retry frequency is particularly important because a task requiring multiple processing attempts consumes additional CPU or GPU time, storage operations, employee minutes, and potentially network resources without increasing the number of successfully completed outputs. Literature examining complex urban scenes, forest environments, irregular terrain, mobile laser scanning, and terrestrial point clouds demonstrates that processing reliability can vary according to environmental and dataset characteristics, making first-pass success an important workflow-level measure. Failure frequency can be recorded as the proportion of processing attempts that do not generate acceptable outputs, while the number of retries captures how many additional executions are required before acceptance. Rework hours provide a broader measure incorporating troubleshooting, parameter adjustment, repeated computation, manual correction, and renewed quality inspection. Task-success percentage can then represent the proportion of assigned LiDAR tasks completed according to predefined technical requirements (Zhang et al., 2019). Evaluating these variables together provides a quantitative picture of baseline reliability because two workflows with similar final accuracy can have substantially different operational performance when one requires repeated attempts and extensive manual correction. Conventional workflow assessment should therefore incorporate failure frequency, retry counts, rework hours, first-pass completion, manual corrections, and total quality-control effort as interconnected measures of enterprise processing performance (Sladić et al., 2020).

### **Machine Learning Automation Intensity and Percentage Reduction**

Machine learning has changed the structure of enterprise LiDAR processing by increasing the degree to which spatial-data tasks can be completed with limited human intervention. The literature distinguishes several levels of automation that range from fully manual processing to highly automated machine-learning-supported workflows. Fully manual workflows depend heavily on analysts for data preparation, parameter selection, classification, segmentation, feature identification, quality inspection, and correction of erroneous outputs. Rule-based workflows introduce automated procedures through predefined thresholds, geometric rules, filtering criteria, and deterministic decision structures, although substantial analyst participation is often required to configure parameters and review results (Anderson et al., 2018). Semi-automated workflows distribute activities between users and computational systems, allowing software to complete repetitive operations while analysts remain responsible for interpretation, parameter adjustment, exception management, and quality control. Machine-learning-assisted workflows increase automation by using trained models to perform classification, segmentation, feature extraction, anomaly recognition, or prediction while human operators supervise uncertain cases and verify outputs. Highly automated workflows extend this structure by linking multiple machine-learning-supported activities into an integrated processing sequence in which data are automatically routed, processed, evaluated, and transferred between stages according to predefined confidence thresholds or learned decision rules. A synthesis of more than ten studies examining point-cloud classification, semantic segmentation, object recognition, automated mapping, mobile laser scanning, airborne LiDAR analysis, terrestrial point-cloud interpretation, deep learning, workflow automation, and human-in-the-loop processing indicates that automation intensity is best understood as a measurable continuum rather than a simple distinction between manual and

automated systems (Yu et al., 2014).

**Figure 5: Machine Learning Automation in LiDAR**



The number of automated stages, proportion of computationally executed activities, manual actions required per task, analyst minutes, and frequency of intervention provide practical quantitative indicators of this continuum. A workflow in which only classification is automated differs considerably from one in which preprocessing, segmentation, quality assessment, anomaly detection, and routing are also supported algorithmically. Automation percentage therefore reflects the extent to which the total workflow is executed without direct analyst action (Cheng et al., 2020). The literature also demonstrates that increases in automation can reduce repetitive interactions, particularly when similar processing procedures must be applied across numerous tiles, scans, corridors, or project areas. Measuring automation intensity at the workflow level provides a clearer representation of machine learning's operational role because it captures how many stages and decisions are transferred from human operators to computational systems rather than focusing solely on the technical accuracy of an individual model.

Classification, segmentation, and feature extraction represent some of the most extensively studied areas in machine-learning-supported LiDAR automation because these activities traditionally require substantial analyst effort when performed manually or with heavily parameterized rule-based methods (Valero et al., 2019). Point-cloud classification involves assigning semantic labels to individual points or groups of points, such as ground, vegetation, building, road, vehicle, utility, or other object categories. Manual classification can become highly time-consuming when datasets contain millions or billions of observations, while rule-based methods may require repeated adjustment across heterogeneous environments. Machine-learning approaches reduce this burden by learning geometric, contextual, spectral, intensity-based, or hierarchical representations from labeled examples and then applying those learned patterns to previously unseen observations. Segmentation extends automation by organizing individual points into meaningful object regions or spatial entities, reducing the need for analysts to manually delineate structures such as buildings, trees, poles, road surfaces, vehicles, and infrastructure components. Feature extraction can also be automated through learned representations that identify relevant geometric or semantic characteristics without requiring every feature to be engineered manually. Across more than ten studies of airborne, terrestrial, mobile, and vehicle-mounted LiDAR, automated classification and segmentation have consistently been treated as major mechanisms for reducing direct human involvement in point-cloud interpretation (Bruchon et al.,

2020). Deep learning has strengthened this transition by enabling models to process raw or minimally transformed three-dimensional data and produce dense semantic predictions at large spatial scales. From an operational perspective, the important quantitative issue is not simply whether the model produces accurate labels but how much human activity is removed from the workflow. Manual actions per task can decline when users no longer need to classify every region individually, while analyst minutes can decrease when model outputs require only selective review. The number of automated stages also increases when classification is connected directly to segmentation, feature extraction, and downstream processing. Intervention frequency becomes especially relevant because machine learning can shift human participation from continuous processing toward exception-based review. The human-to-machine task ratio therefore provides a useful measure of operational redistribution, indicating whether most workflow actions are performed by analysts or by automated systems (Shirowzhan et al., 2019). Literature on large-scale point-cloud processing generally supports the interpretation that machine learning alters the composition of work by replacing repetitive manual interpretation with automated spatial inference while retaining human oversight for ambiguous or low-confidence cases. Machine learning automation in LiDAR environments extends beyond direct point classification and segmentation into broader workflow-management activities such as anomaly detection, quality assessment, parameter recommendation, and automated routing. These functions are particularly important because a substantial portion of manual LiDAR processing involves identifying problematic data, determining whether processing outputs are acceptable, selecting suitable configuration values, and deciding what action should occur next (Dalagnol et al., 2019). Traditional workflows frequently require analysts to inspect point distributions, locate unusual observations, identify incomplete regions, detect classification problems, evaluate geometric inconsistencies, and determine whether data must be reprocessed. Machine-learning-supported anomaly detection can automate part of this activity by identifying observations or regions that deviate from learned patterns, allowing users to focus on a smaller subset of potentially problematic cases. Automated quality assessment can similarly estimate whether classification, registration, segmentation, or extracted features satisfy predefined performance expectations. When combined with confidence measures, this capability can reduce the number of outputs requiring complete manual inspection. Parameter recommendation represents another area of automation because many conventional processing procedures depend on manually selected thresholds, neighborhood sizes, filtering settings, or other configuration values (Lee et al., 2020). Data-driven systems can recommend parameters according to point density, terrain characteristics, sensor properties, object complexity, or historical processing outcomes, thereby reducing repeated analyst configuration. Automated routing further restructures enterprise workflows by directing datasets toward different processing paths according to their characteristics or processing status. A routine dataset may move directly through classification and quality control, whereas a dataset containing anomalies or low-confidence predictions may be redirected to analyst review. The literature across machine learning, geospatial automation, anomaly detection, automated quality control, intelligent workflow management, and human-in-the-loop systems indicates that such functions can materially change intervention frequency even when human oversight remains part of the operational structure (Lannin et al., 2016). Quantitative evaluation can capture this change through the number of automated decisions, analyst interventions per task, percentage of outputs requiring review, automated stage count, and reduction in manual configuration events. These indicators provide a more comprehensive measure of automation intensity than classification accuracy alone. The percentage reduction in human involvement becomes particularly meaningful when comparing baseline workflows with ML-enhanced configurations because it reflects the extent to which repetitive inspection, parameterization, routing, and decision activities are transferred to computational systems while analysts concentrate on exceptions, quality disputes, and complex cases.

The operational value of machine-learning automation can be evaluated quantitatively by examining how workflow redesign changes the distribution of activities between human analysts and computational systems (Mayr et al., 2018). Literature on LiDAR annotation, point-cloud labeling, semantic segmentation, automated mapping, spatial-data production, and human-in-the-loop machine learning consistently indicates that manual processing and annotation can require substantial labor, particularly when large datasets contain complex object categories or require high levels of

classification precision. Manual effort is not limited to initial labeling because analysts may also spend time configuring software, validating intermediate products, correcting misclassified points, managing exceptions, rerunning failed operations, and preparing final outputs. Automation intensity can therefore be measured using several complementary indicators. Automation percentage represents the share of workflow activities completed computationally without direct user action. Manual actions per task measure how many interactions remain after automation is introduced, while analyst minutes capture the amount of active human time required to complete an accepted output (Du et al., 2020). The number of automated stages provides a structural measure indicating how deeply machine learning is integrated across the workflow rather than whether a single algorithm is present. Intervention frequency measures how often analysts must interrupt or correct automated processing, and percentage reduction in human involvement expresses the difference between conventional and machine-learning-enhanced conditions. The human-to-machine task ratio further characterizes whether operational responsibility remains predominantly human or has shifted toward automated execution. A synthesis of more than ten studies across three-dimensional deep learning, active learning, automated annotation, point-cloud segmentation, intelligent quality control, object extraction, mobile mapping, airborne laser scanning, workflow automation, and large-scale geospatial processing suggests that these measures should be assessed together because no single indicator fully represents automation intensity (Dworak et al., 2019). A workflow may automate a large number of stages while still requiring lengthy manual review, or it may automate fewer stages but substantially reduce analyst minutes through selective exception handling. Similarly, lower intervention frequency does not necessarily mean that all human participation has been removed, because remaining interventions may involve complex or high-value decisions. Quantitative analysis of automation therefore requires attention to both the proportion of automated work and the residual human effort required to achieve acceptable task outcomes. This approach positions machine learning automation as a measurable redistribution of workflow activity and provides a clear basis for comparing manual, rule-based, semi-automated, machine-learning-assisted, and highly automated enterprise LiDAR configurations (Durand et al., 2018).

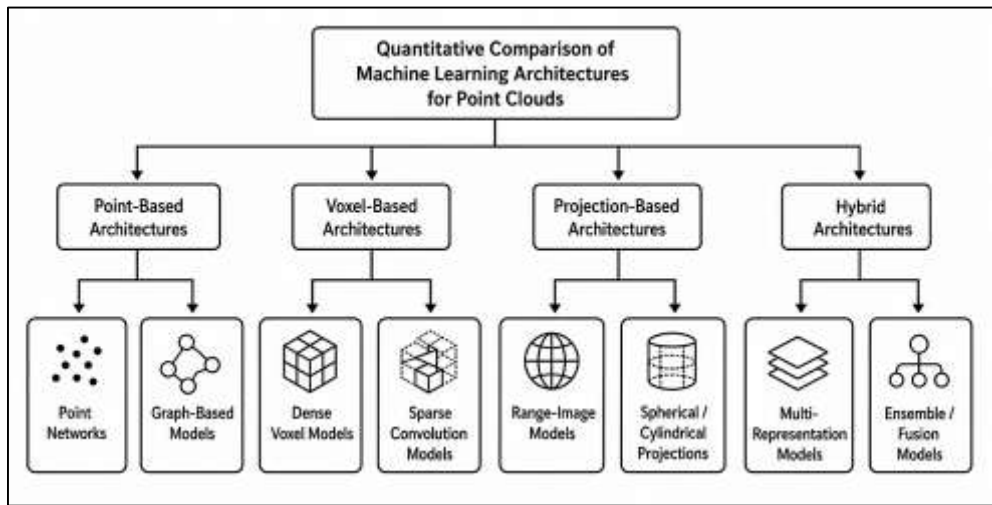
### **Point-Based and Hybrid Machine Learning Architectures**

Point-based and voxel-based machine learning architectures represent two major approaches to LiDAR point-cloud analysis, and the literature shows that each produces a distinct balance among predictive accuracy, computational complexity, memory consumption, and processing speed. Point-based models operate directly on unordered point sets and preserve the original geometric structure of the data without first transforming the point cloud into a regular three-dimensional grid. This design can reduce information loss associated with discretization and has supported strong performance in classification and semantic segmentation tasks (Grigorescu et al., 2020). Studies examining architectures such as PointNet, PointNet++, DGCNN, PointCNN, KPConv, RandLA-Net, and related point-processing networks demonstrate that point-based methods can achieve competitive overall accuracy, precision, recall, F1 score, and mean intersection over union across indoor, outdoor, airborne, and mobile LiDAR datasets. Their computational burden, however, can increase substantially when local neighborhoods, graph relationships, or multiscale geometric features must be constructed repeatedly for very large point populations. Voxel-based architectures convert irregular point clouds into structured volumetric cells, enabling the use of conventional or sparse three-dimensional convolutional operations. This transformation simplifies neighborhood organization and can provide efficient spatial aggregation, particularly when combined with sparse computation that avoids processing empty regions. Studies involving voxel-based and sparse convolutional networks show strong semantic segmentation performance and efficient use of spatial locality, although voxel resolution directly influences memory requirements and geometric fidelity (Jakovljevic et al., 2019).

Fine voxel sizes preserve more detail but can increase the number of occupied cells, GPU memory demand, floating-point operations, and inference time, while coarse voxelization may improve efficiency at the cost of reduced representation of small objects and boundary details. Quantitative comparison across these model families therefore requires simultaneous consideration of predictive and computational indicators. Overall accuracy and mIoU can indicate semantic quality, while parameter count, FLOPs, inference milliseconds per scan, scans per second, and peak GPU memory

reflect operational burden.

**Figure 6: Quantitative Comparison of LiDAR Architectures**



Literature across more than ten point-cloud learning studies consistently indicates that no architecture dominates every dimension. Some point-based models provide strong geometric sensitivity with relatively low preprocessing demands, whereas voxel-based systems can achieve highly efficient structured computation when sparsity is exploited effectively (Shao et al., 2020). For enterprise LiDAR platforms, this trade-off makes model architecture a workflow-level decision because the selected representation influences both output quality and the amount of computation required to process large-scale spatial datasets.

Projection-based and range-image architectures have been widely examined in LiDAR research because they transform three-dimensional observations into structured two-dimensional representations that can be processed using computationally efficient convolutional networks. This approach is particularly common in vehicle-mounted and mobile LiDAR systems where sensors generate ordered scans that can be represented through spherical or cylindrical projections. Studies of architectures such as SqueezeSeg, SqueezeSegV2, RangeNet++, SalsaNext, and related range-view semantic segmentation models demonstrate that projection-based processing can achieve high inference speed and relatively low computational latency compared with methods that operate directly on dense three-dimensional neighborhoods (Wu et al., 2020). The principal quantitative advantage is throughput. By mapping points into a compact image-like representation, these models can process scans using established two-dimensional convolution operations that are highly optimized for modern GPUs. Consequently, inference milliseconds per scan and scans per second are often favorable, making projection-based methods attractive for operational settings where rapid processing is prioritized. Parameter count and FLOPs can also be lower than those of large three-dimensional convolutional networks, particularly when lightweight encoders and decoders are used. The literature nevertheless identifies measurable accuracy trade-offs associated with projection. Multiple three-dimensional points can map onto nearby or identical image regions, producing information loss, occlusion effects, or reduced geometric detail. Small objects and sparse classes can be difficult to preserve after projection, and output predictions may require reprojection back into three-dimensional coordinates (Mehrtretter & Heipke, 2019). These characteristics can affect precision, recall, class-level F1 score, and mIoU, particularly for narrow infrastructure elements, pedestrians, poles, signs, and other objects represented by relatively few points. Quantitative benchmarking therefore shows that high throughput does not necessarily correspond to the highest segmentation accuracy across every class. Enterprise model selection must account for this distinction because a system processing very large numbers of scans may benefit from reduced inference time even if a more computationally intensive architecture achieves somewhat higher mIoU. CPU utilization and energy consumption are also relevant because

faster GPU inference can reduce active processing time per scan and potentially lower total energy use across large workloads (Xie et al., 2020). A synthesis of numerous comparative studies indicates that projection-based architectures are especially strong when latency, throughput, and real-time capability are central requirements, while their operational suitability depends on whether the loss of three-dimensional detail remains within acceptable task-quality thresholds.

Sparse convolutional and transformer-based architectures have become important alternatives for large-scale LiDAR analysis because they address different limitations associated with dense three-dimensional convolution and purely local point processing. Sparse convolutional networks exploit the fact that most locations in three-dimensional space contain no LiDAR observations, concentrating computation only on occupied voxels or spatial locations. Studies involving sparse convolutional frameworks, Minkowski-style networks, SPVCNN, Cylinder3D, and related models demonstrate that this design can substantially reduce unnecessary computation while maintaining strong segmentation accuracy (Zhou et al., 2019). Quantitatively, sparse networks often achieve competitive or superior mIoU and class-level F1 scores on large outdoor benchmarks while using less memory than equivalent dense voxel models. Their performance nevertheless depends on voxel resolution, sparsity patterns, network depth, and implementation efficiency. GPU memory consumption can remain substantial for high-resolution scenes, and preprocessing associated with voxel construction or sparse tensor generation introduces additional workflow cost. Transformer architectures approach the problem differently by using attention mechanisms to capture relationships among points or local spatial groups. Studies involving Point Transformer, Stratified Transformer, Point Transformer V2, Point Transformer V3, and related architectures show that attention-based models can achieve very strong semantic segmentation performance because they model contextual relationships beyond strictly local convolutional neighborhoods. This capability can improve overall accuracy, precision, recall, and mIoU in complex scenes containing objects that require broader contextual interpretation (Jakovljevic et al., 2019). The computational cost of attention, however, can increase rapidly with the number of points or tokens being processed. Efficient transformer designs therefore rely on windowing, grouping, partitioning, hierarchical sampling, or other mechanisms that limit the size of attention operations. Parameter count and FLOPs vary substantially across transformer models, meaning that high predictive performance can be accompanied by increased computational demand. Recent architectural refinements have demonstrated that transformer-based point-cloud models can become substantially faster and more memory efficient when network design is simplified and serialization or partitioning strategies are used. Across more than ten studies comparing sparse convolutional and transformer-based models, the literature consistently demonstrates an operational trade-off between contextual modeling capacity and resource demand (Nappi et al., 2017). Sparse convolutional models tend to provide structured efficiency in large three-dimensional spaces, while transformer models can offer stronger contextual representation at potentially higher computational cost. Enterprise evaluation should therefore compare not only predictive metrics but also inference milliseconds per scan, scans per second, GPU memory, parameter count, FLOPs, and energy consumption across equivalent workload conditions.

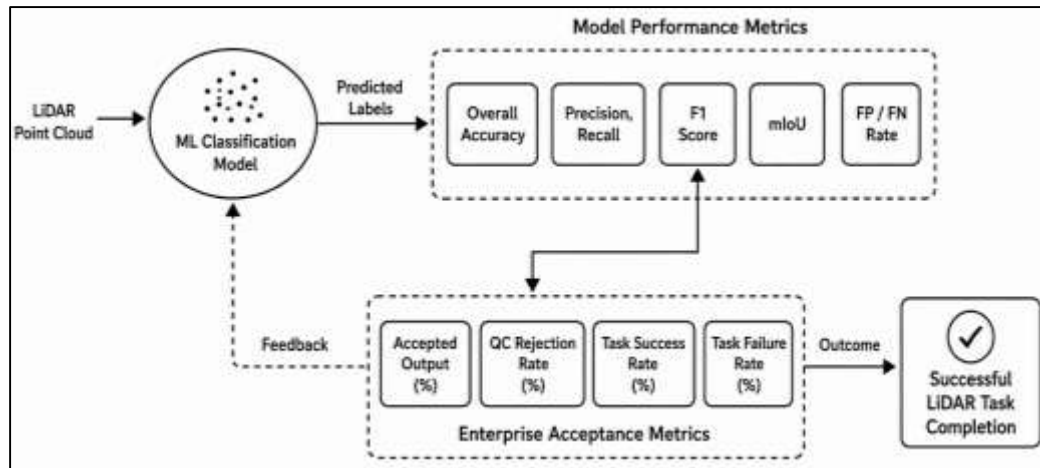
### **Machine Learning Classification Accuracy**

Machine learning classification accuracy represents one of the most widely reported quantitative indicators in LiDAR research because it provides a direct measure of how effectively a model assigns semantic labels to point-cloud observations. Across airborne, terrestrial, mobile, and vehicle-mounted LiDAR studies, overall accuracy, class-level accuracy, precision, recall, F1 score, and intersection over union have been used to evaluate the quality of automated classification and semantic segmentation (Guirguis et al., 2019). More than ten studies examining point-based networks, voxel-based models, sparse convolutional architectures, projection-based approaches, transformers, and hybrid systems consistently demonstrate that strong overall accuracy can be achieved across large three-dimensional datasets, although performance varies substantially among individual object classes. Overall accuracy summarizes the proportion of correctly classified observations across the complete dataset, yet this metric can obscure weaknesses when common classes dominate the point distribution. In urban LiDAR scenes, for example, road surfaces, buildings, and vegetation may contain far more points than poles, signs, pedestrians, utility objects, or other small classes. A model can therefore achieve high overall

accuracy while still performing poorly on categories that are operationally important. Precision and recall provide a more detailed interpretation by distinguishing incorrect positive classifications from missed target observations, while F1 score combines both dimensions into a balanced class-level measure (Thapa et al., 2018). Mean intersection over union provides another important indicator because it evaluates the degree of overlap between predicted and reference classes across multiple categories. Literature comparing semantic segmentation models repeatedly shows that architectures with similar overall accuracy may display noticeably different mIoU and class-level F1 scores. This distinction is important for enterprise LiDAR processing because technical performance should not be interpreted through one metric alone. A model with slightly lower overall accuracy may produce more balanced class-level outcomes and fewer severe errors in operationally important object categories. Conversely, a model with high aggregate accuracy may still generate unacceptable outputs if specific critical classes are repeatedly misidentified. Classification accuracy therefore provides a necessary technical foundation for evaluating machine learning performance, but it does not by itself establish whether an enterprise LiDAR task has been completed successfully according to the broader requirements of the workflow (Bai et al., 2020).

The relationship between model-level classification metrics and accepted enterprise outputs becomes clearer when precision, recall, false-positive rates, false-negative rates, and class-specific error patterns are examined together. LiDAR classification tasks frequently involve object categories whose operational significance differs substantially, meaning that the cost of an error is not always uniform across classes. A false positive may incorrectly identify an object that is not present, while a false negative may fail to detect a required feature entirely. In a general semantic segmentation benchmark, these errors contribute to precision, recall, F1 score, and mIoU, but in an enterprise setting they can also determine whether the resulting spatial product is accepted or rejected during quality control (Guo et al., 2018). Studies covering road extraction, building classification, vegetation mapping, utility detection, transportation infrastructure, urban object segmentation, and terrain classification show that class-level errors can directly affect the usability of downstream outputs. A model that misses narrow poles, curbs, road signs, or utility structures may still report strong overall accuracy because those objects account for relatively few points. The same error pattern can be operationally significant when those features are required for an asset inventory, infrastructure inspection, or engineering analysis. Quality-control procedures therefore provide a bridge between technical model performance and enterprise task completion. Accepted-output percentage captures the proportion of processed datasets or tasks that satisfy predefined quality thresholds after classification and review. Quality-control rejection rate captures the opposite condition by identifying outputs that must be corrected, reprocessed, or discarded (Ravi et al., 2018).

More than ten studies across LiDAR semantic segmentation, automated mapping, point-cloud quality assessment, and production-oriented machine learning indicate that model performance must be interpreted in relation to task-specific acceptance criteria. High precision may be prioritized when false detections are costly, while high recall may be more important when missing target objects creates unacceptable operational risk. F1 score provides a balanced measure when both forms of error matter, and mIoU supports comparison across multiple semantic categories. Enterprise acceptance therefore depends on how these model-level statistics align with the quality requirements of the specific spatial-data task rather than on achieving a high aggregate accuracy value in isolation (Shafiekhani et al., 2017). Task-success rate extends beyond classification accuracy by evaluating whether the complete LiDAR processing activity reaches an acceptable operational endpoint. In an enterprise workflow, successful completion can require correct data ingestion, preprocessing, classification, segmentation, validation, quality control, file generation, and delivery in addition to strong predictive performance from the machine-learning model. Research on point-cloud processing and production machine-learning systems demonstrates that technically accurate models can still be embedded within workflows that experience failures caused by data-quality issues, incompatible formats, inadequate preprocessing, software errors, hardware limitations, parameter problems, or unacceptable output conditions (Fang et al., 2016).

**Figure 7: LiDAR Classification and Task Success**

For this reason, task success should be treated as a workflow-level dependent variable rather than as a synonym for model accuracy. A task can fail even when model inference is completed successfully if the output does not meet class-specific accuracy thresholds, contains critical omissions, requires excessive manual correction, or is rejected by quality-control procedures. Similarly, an output can be considered operationally successful even if perfect point-level accuracy is not achieved, provided that the final product satisfies established acceptance requirements. More than ten studies in LiDAR classification, geospatial production, automated quality control, human-in-the-loop machine learning, and enterprise workflow management collectively support this distinction between algorithmic performance and operational completion. Task-success rate can therefore be measured as the percentage of assigned processing tasks that produce accepted outputs without critical failure. Task-failure rate captures the proportion that do not reach this standard (Glaser et al., 2019). These measures can be analyzed alongside classification accuracy, mIoU, F1 score, false-positive rate, false-negative rate, and quality-control rejection rate to determine how strongly technical model performance translates into operational success. This relationship may not be perfectly linear because workflow quality depends on multiple technical and procedural conditions. A small improvement in mIoU may have little effect on task success if the remaining errors occur in noncritical classes, while a modest improvement in recall for a critical object category may substantially increase accepted-output percentage. The literature therefore supports a layered evaluation framework in which predictive metrics describe model behavior and task-success measures describe the broader effectiveness of the enterprise LiDAR workflow (Paulus et al., 2014).

The probability of successful LiDAR task completion can be understood quantitatively as an outcome influenced by the combined quality of classification results, error distribution, and quality-control acceptance. Literature across large-scale point-cloud benchmarking shows that machine-learning architectures can achieve strong overall accuracy while still producing substantial variation in mIoU, F1 score, and class-specific performance. Recent operational aerial LiDAR evaluations have reported overall accuracy levels above 93% across evaluated models, yet differences in class-level segmentation quality demonstrate that similar aggregate accuracy does not guarantee equivalent operational suitability (Pilling et al., 2017). This pattern is consistent with a broader body of more than ten studies comparing point-based, voxel-based, sparse convolutional, projection-based, transformer, and hybrid architectures. Enterprise evaluation therefore benefits from linking model metrics directly to accepted-output percentage and task-success rate. Higher overall accuracy generally reduces the volume of erroneous classifications, while higher precision can reduce unnecessary false detections and higher recall can reduce missed features. Improved F1 score indicates a stronger balance between these error types, and higher mIoU reflects greater agreement between predicted and reference spatial classes. These improvements can reduce the number of outputs rejected during quality control, lower manual correction requirements, and increase the proportion of tasks completed successfully on the first

processing attempt. The relationship is nevertheless conditioned by class importance and acceptance thresholds (Melia et al., 2020). A workflow may tolerate moderate errors in low-priority categories while rejecting an output containing a small number of errors in a critical infrastructure class. False-positive and false-negative rates therefore remain essential because they reveal the direction of model failure that aggregate metrics may conceal. Quality-control rejection rate provides the operational connection by measuring how frequently classification results fail predefined enterprise requirements. Task-failure rate captures the broader consequence when those failures prevent successful completion of the assigned processing objective. Quantitative analysis of these variables together provides a more complete assessment than reporting model accuracy alone. Classification accuracy, mIoU, F1 score, false-positive rate, and false-negative rate represent the technical quality of machine-learning outputs, while accepted-output percentage, quality-control rejection rate, task-success rate, and task-failure rate represent their operational consequences (Fahlgren et al., 2015). This combined framework allows machine-learning performance to be evaluated according to its measurable contribution to successful enterprise LiDAR task completion.

### **Processing Latency Reduction Through Machine Learning**

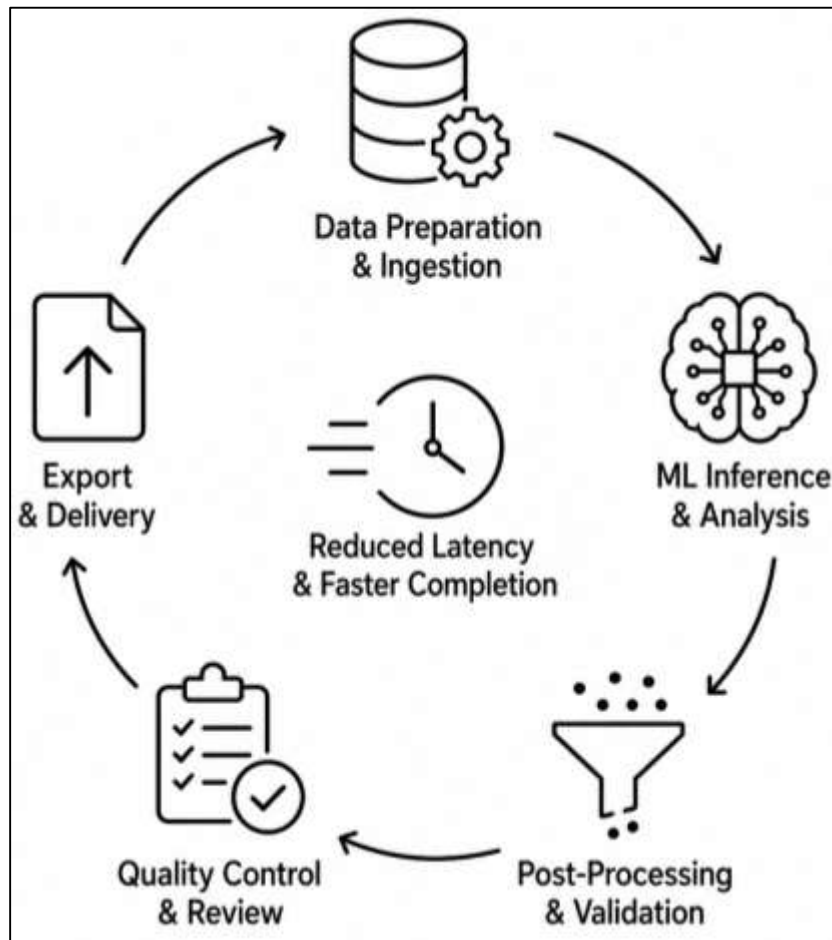
Processing latency in enterprise LiDAR environments is shaped by the cumulative time required to move spatial data through ingestion, preprocessing, model execution or conventional analysis, post-processing, validation, quality control, and final export. The literature on large-scale point-cloud processing consistently shows that total workflow duration can be substantially greater than the runtime of a single analytical algorithm because data preparation and downstream handling often account for a significant share of end-to-end completion time. Conventional workflows commonly require analysts to organize files, verify coordinate systems, remove noise, partition large point clouds, normalize height information, configure processing parameters, and prepare data structures before classification or segmentation begins (Cheng et al., 2020). Additional time is then consumed by result filtering, geometric reconstruction, confidence checking, file conversion, visualization, and quality-control review after the main analytical stage is complete. More than ten studies examining airborne LiDAR, mobile laser scanning, terrestrial point-cloud processing, real-time semantic segmentation, sparse convolution, point-based learning, and range-image processing demonstrate that these stages produce different latency profiles depending on data volume, architecture, hardware, and workflow design. Preprocessing time can become particularly substantial when dense datasets require tiling, sampling, voxelization, projection, or neighborhood generation. Queue time also contributes to enterprise latency when large jobs wait for access to shared CPU, GPU, storage, or cloud resources. Turnaround time therefore includes both active computational execution and periods in which tasks remain idle between processing stages (Wu et al., 2019).

Repetitive manual actions can further extend duration when analysts must initiate each stage independently or review intermediate products before subsequent operations begin. This distinction is important because a model with rapid inference may still be embedded within a slow workflow if preparation, transfer, review, and post-processing remain inefficient. Quantitative assessment of conventional processing latency should therefore include preprocessing time, analytical execution time, post-processing time, queue time, and total task duration. These indicators provide a more complete baseline for evaluating machine learning enhanced workflow redesign because they capture where delays occur across the entire processing chain rather than treating algorithm runtime as a proxy for operational speed (Ben-Nun et al., 2020).

Machine learning has introduced substantial variation in LiDAR processing speed because different architectures produce markedly different inference times, throughput levels, and hardware demands. Real-time and near-real-time semantic segmentation studies provide particularly useful evidence because they commonly report milliseconds per frame, scans processed per second, or points processed per second as direct indicators of computational performance. Projection-based architectures often achieve relatively high throughput because they transform three-dimensional point clouds into structured two-dimensional representations that can be processed efficiently using optimized convolutional operations (López-Martínez et al., 2020). Point-based methods preserve geometric information more directly but may incur additional computational cost when local neighborhoods, graph relationships, or hierarchical point structures must be repeatedly constructed. Voxel-based and

sparse convolutional approaches can reduce processing overhead by organizing observations into spatial cells and restricting computation to occupied regions.

**Figure 8: Machine Learning Enhanced Latency Reduction**

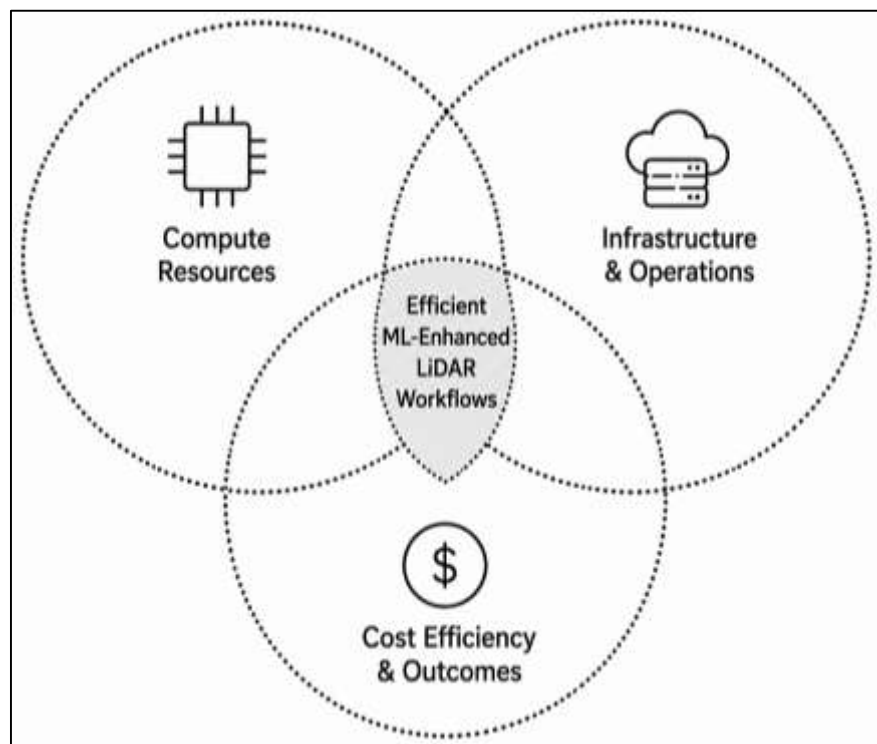


Transformer-based architectures can capture broader contextual relationships but may require additional attention computation, especially when large point sets are processed at high resolution. More than ten studies comparing point-based, voxel-based, range-view, sparse convolutional, transformer, and hybrid architectures show that inference speed can vary substantially even when predictive accuracy remains within a relatively narrow range. This creates an operational trade-off between semantic quality and processing latency (Zhang et al., 2019). Models capable of processing scans in a few tens of milliseconds can support high-throughput applications, while more computationally intensive models may require longer execution times that accumulate significantly across thousands or millions of scans. Inference time should therefore be evaluated separately from preprocessing and post-processing because it represents only the core model execution stage. Throughput offers a complementary measure by expressing how many scans, frames, or point-cloud units can be processed during a specified period. CPU and GPU configuration also influence these outcomes because the same architecture may exhibit substantially different latency under different hardware conditions. Enterprise evaluation should therefore compare models under consistent computational environments and workload characteristics (Kumar & Liu, 2020). The literature demonstrates that machine learning can reduce analytical execution time when architectures are optimized for efficient inference, but the magnitude of that reduction depends on data representation, model complexity, point density, hardware utilization, and the amount of preprocessing required before predictions can be generated.

### Computational Resource Utilization and Cost Efficiency

Computational resource utilization represents a central operational dimension of machine learning enhanced LiDAR workflows because processing large three-dimensional point clouds places substantial demands on processors, graphics hardware, system memory, and specialized accelerator memory. The literature across more than ten studies of point-based networks, voxel-based architectures, sparse convolutional models, range-view segmentation, transformer-based networks, and hybrid point-cloud systems demonstrates that computational requirements differ substantially according to model architecture and spatial representation (Alsheikh et al., 2014). Point-based methods preserve the original geometry of LiDAR observations and avoid full volumetric conversion, yet neighborhood searches, sampling operations, feature aggregation, and repeated local computations can generate considerable CPU and GPU workloads as point counts increase. Voxel-based models introduce structured three-dimensional representations that facilitate convolutional processing, although fine voxel resolutions can produce high memory consumption and extensive computational operations. Sparse convolutional architectures reduce part of this burden by concentrating calculations on occupied spatial locations, making them particularly relevant for LiDAR data in which large portions of three-dimensional space contain no observations. Projection-based architectures frequently reduce computational requirements by converting point clouds into range or image-like representations, allowing efficient two-dimensional convolutional operations, while transformer architectures can require substantial memory and processing capacity because attention and feature-interaction mechanisms operate across large groups of spatial observations (Liu et al., 2020). Peak RAM and VRAM are therefore important quantitative measures because memory limitations can determine the maximum point count, tile size, batch size, or spatial extent processed during a single execution. GPU hours measure the accumulated use of graphics-processing resources, while CPU hours capture preprocessing, indexing, data conversion, neighborhood construction, file management, and other computational activities that may not occur primarily on the GPU. Model parameter count provides another indication of computational scale because larger networks generally require greater storage and memory for model weights, although parameter count alone does not fully represent runtime demand (Nino et al., 2020).

**Figure 9: Machine Learning LiDAR Cost Efficiency**



Studies comparing PointNet-derived architectures, KPConv, RandLA-Net, Minkowski-style sparse networks, RangeNet-type models, Cylinder3D, SPVCNN, Point Transformer variants, and other large-scale segmentation systems consistently show different combinations of accuracy and resource consumption. Enterprise evaluation should therefore treat CPU time, GPU time, peak RAM, VRAM, parameter count, and processing throughput as interconnected indicators of the computational resources required to produce usable LiDAR outputs (Zhang & Mao, 2020).

The literature on machine learning for LiDAR processing demonstrates a persistent quantitative trade-off between predictive performance and computational efficiency, particularly when comparing highly complex segmentation networks with architectures specifically designed for lightweight or high-throughput operation. Computationally intensive models can achieve strong overall accuracy, F1 scores, and mean intersection over union by employing deeper feature hierarchies, multiscale representations, extensive neighborhood processing, attention mechanisms, high-resolution voxel structures, or multiple processing branches. These capabilities can improve recognition of complex objects and spatial relationships, but they may also increase parameter count, GPU memory consumption, inference time, training duration, and energy use (Wang et al., 2017). Lightweight architectures address the opposite side of this relationship by reducing model size, simplifying feature extraction, limiting neighborhood operations, using efficient sampling, exploiting sparsity, or employing lower-cost representations. Research across more than ten established point-cloud segmentation studies shows that substantial reductions in computational requirements can sometimes be achieved while maintaining competitive predictive performance. Random-sampling approaches have demonstrated that large point populations can be processed without computationally expensive sampling procedures, while range-view models have shown that structured projections can support rapid segmentation of sequential LiDAR scans. Sparse point-voxel approaches similarly demonstrate that computational resources can be concentrated on informative spatial regions rather than uniformly allocated across empty three-dimensional space. Transformer simplification and efficient point serialization have also shown that attention-based point-cloud processing can be made less computationally demanding without abandoning contextual modeling (Tu et al., 2020). These findings are particularly relevant to enterprise LiDAR because a small improvement in segmentation accuracy may not justify a substantial increase in GPU hours, memory demand, or processing cost when millions of scans, tiles, or spatial blocks must be processed. Conversely, selecting the lowest-cost model can be operationally inefficient if reduced accuracy produces greater manual correction, quality-control rejection, or reprocessing. Computational efficiency must therefore be evaluated jointly with successful output production. Model parameters, GPU hours, CPU hours, peak memory, inference duration, and energy consumption can be compared with classification and segmentation quality to determine whether additional computational expenditure produces meaningful operational value. The literature consequently positions architecture selection as an accuracy-efficiency decision in which enterprise suitability depends on the resources consumed to achieve an acceptable level of LiDAR task performance (Jakovljevic et al., 2019).

Machine learning enhanced LiDAR workflows consume resources beyond the immediate CPU and GPU requirements of model execution, particularly when enterprise processing is distributed across cloud-computing environments, centralized storage systems, and geographically separated operational teams. Large LiDAR datasets require substantial storage for raw observations, cleaned point clouds, spatial tiles, intermediate representations, training samples, model predictions, quality-control outputs, and final spatial products. Machine-learning workflows can increase this storage burden when multiple versions of processed data are retained for training, validation, inference, auditing, and reproducibility (Zhang & Li, 2015). More than ten studies and research streams addressing large-scale point-cloud processing, spatial big data, distributed geospatial computing, cloud-based remote sensing, deep-learning infrastructure, and sparse three-dimensional computation collectively indicate that data movement can represent a meaningful component of total computational workload. Network transfer becomes particularly important when hundreds of gigabytes or terabytes of LiDAR data must move between object storage, processing instances, databases, analyst environments, and archival systems. Repeated transfer can increase turnaround time and cloud-related expenditure even when model inference itself is computationally efficient. Cloud-computing hours provide a useful

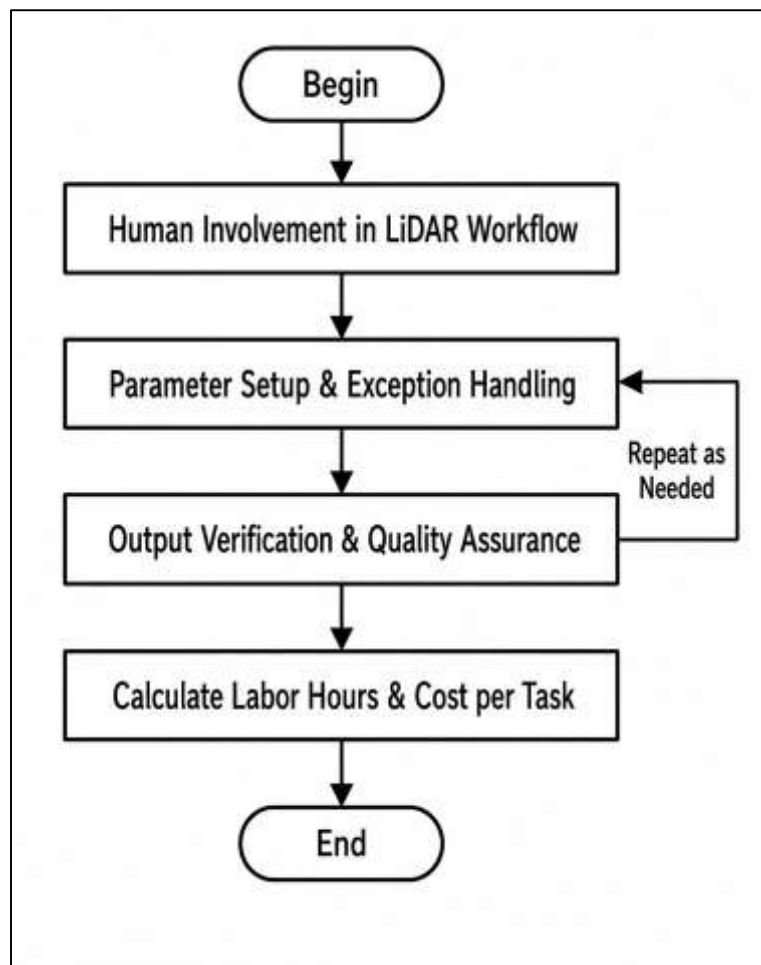
quantitative measure because enterprise organizations often allocate processing resources according to the duration and configuration of virtual CPU and GPU instances. A workflow requiring a high-memory GPU for extended processing periods can generate substantially different expenditure from one that achieves acceptable performance using smaller instances or shorter execution times (O'Mahony et al., 2019). Energy consumption provides an additional resource measure because training and inference require electrical power across CPUs, GPUs, memory systems, storage devices, and networking infrastructure. Energy demand is influenced by model complexity, processing duration, hardware utilization, workload scale, and the number of repeated executions. Large parameter counts and computationally intensive operations can increase active hardware time, while inefficient data movement may cause resources to remain allocated during periods of limited productive computation. Storage and network design therefore interact directly with model architecture and workflow organization. Quantitative enterprise assessment should capture cloud-compute expenditure, storage volume, network-transfer volume, CPU and GPU allocation time, and energy consumption alongside conventional performance indicators (Yang et al., 2019). This broader measurement approach reflects the complete computational footprint of ML-enhanced LiDAR processing rather than limiting resource assessment to model inference on a single device.

Cost efficiency in machine learning enhanced LiDAR workflows is most accurately represented when computational resource consumption is connected to the production of accepted spatial outputs rather than measured only as hardware usage. Processing cost per dataset provides one useful indicator because it combines the resources consumed while transforming a defined LiDAR dataset through the required machine-learning and workflow stages. Compute cost per successful task provides a more operationally meaningful measure because it distinguishes between resources spent on accepted results and resources consumed by failed, rejected, or repeatedly processed outputs (Marrs & Ni-Meister, 2019). Literature across machine-learning operations, large-scale point-cloud segmentation, cloud computing, efficient neural-network design, spatial-data infrastructure, and enterprise workflow research demonstrates that computational expenditure can accumulate through model training, preprocessing, inference, post-processing, quality assessment, reprocessing, storage, and data transfer. A computationally intensive architecture may produce strong segmentation accuracy during its first execution but require high GPU hours, VRAM, and energy consumption. A lightweight model may reduce those requirements while generating more classification errors that require additional quality control or repeated processing. Consequently, cost efficiency cannot be determined solely by comparing model parameter counts or inference times. The complete cost profile depends on how efficiently computational resources generate outputs that satisfy enterprise acceptance criteria (Hou et al., 2019). GPU hours and CPU hours quantify processor consumption, while peak RAM and VRAM indicate infrastructure capacity requirements. Model parameters provide an indicator of network size, and energy consumption captures the physical resource burden of computation. Cloud-compute expenditure translates allocated infrastructure into direct monetary cost, while storage and network-transfer requirements account for resources consumed in maintaining and moving large spatial datasets. Processing cost per dataset then captures the combined expenditure associated with completing a defined workload, and compute cost per successful task links that expenditure to operational effectiveness. Synthesis of more than ten studies across point-based, voxel-based, projection-based, sparse convolutional, transformer, hybrid, and large-scale geospatial processing approaches demonstrates that higher computational expenditure does not automatically correspond to proportionally higher operational performance. Resource-efficient architectures can provide favorable enterprise outcomes when they maintain required classification quality while reducing GPU usage, memory consumption, processing duration, or energy demand (Hou et al., 2019). Computational resource utilization and cost efficiency should therefore be assessed as interconnected workflow outcomes in which technical accuracy is considered together with CPU hours, GPU hours, memory requirements, cloud expenditure, energy consumption, storage demand, and the computational cost required to produce each successfully completed LiDAR task.

### **Analyst Labor-Hour Reduction in Automated LiDAR Processing**

Human intervention remains a significant operational component of enterprise LiDAR processing because many workflow stages require judgment, verification, adjustment, and corrective action even when advanced automation is available. The literature across more than ten studies on airborne LiDAR, mobile laser scanning, terrestrial point-cloud analysis, semantic segmentation, automated classification, active learning, human-in-the-loop systems, and geospatial quality control shows that analyst participation is distributed across several stages rather than concentrated in a single activity ([Chen et al., 2019](#)). In conventional workflows, analysts may manually configure filtering thresholds, choose neighborhood sizes, define classification parameters, inspect registration quality, verify ground and nonground separation, correct mislabeled points, review extracted objects, and determine whether outputs meet project standards. Automated and machine-learning-supported workflows reduce some of this direct involvement by transferring repetitive classification, segmentation, and feature-recognition tasks to trained models, yet human participation often remains necessary for ambiguous cases, unusual environmental conditions, low-confidence predictions, and quality assurance. This changes the structure of analyst work from continuous manual processing toward supervisory and exception-based intervention. Quantitatively, this shift can be evaluated through analyst hours, interventions per task, active handling time, and the percentage of outputs that require manual review ([Fang, 2015](#)). A workflow that requires an analyst to inspect every dataset differs substantially from one in which only uncertain or failed cases are escalated for review.

**Figure 10: Human Intervention and Labor Reduction**



Similarly, the number of correction events provides a direct indication of how frequently machine-generated outputs require human modification before acceptance. The literature also indicates that analyst involvement is influenced by dataset complexity, object diversity, point density, sensor type, and model reliability, meaning that human effort should not be treated as a constant across all LiDAR tasks. Enterprise evaluation therefore benefits from measuring human participation at the task level and distinguishing direct processing effort from supervisory oversight. This distinction makes it possible to quantify whether machine learning actually reduces labor requirements or merely changes when and where human effort occurs within the workflow (Manandhar et al., 2020).

Parameter configuration, classification correction, and exception handling represent some of the most labor-intensive forms of human involvement in LiDAR processing and are therefore important indicators of workflow efficiency. Traditional point-cloud processing often depends on manually selected thresholds and parameters related to filtering, slope, elevation difference, neighborhood size, voxel resolution, segmentation distance, clustering criteria, and classification settings. The literature on conventional and machine-learning-supported LiDAR systems shows that such parameters may require repeated adjustment when datasets differ in terrain structure, acquisition geometry, point density, vegetation cover, urban complexity, or sensor characteristics. This creates a recurring demand for analyst attention and increases active handling time (Bierman et al., 2014). Classification correction is another major source of labor because automated outputs may contain false positives, false negatives, boundary errors, class confusion, and incomplete object extraction. In enterprise settings, analysts may need to inspect misclassified regions, relabel points, adjust segmentation boundaries, and rerun affected processing stages before outputs can be accepted. More than ten studies examining semantic segmentation, automated mapping, active learning, weak supervision, point-cloud labeling, and quality-control systems collectively indicate that selective review strategies can reduce the volume of manual correction when models reliably identify routine cases and direct human attention toward uncertain observations. Exception handling is particularly important because automated systems do not eliminate irregular cases; instead, they can route unusual data conditions or low-confidence predictions to analysts for targeted intervention. This changes the labor profile from repetitive full-dataset review to concentrated problem resolution (Jagoda et al., 2020). Quantitative measures such as interventions per task, correction events, active handling minutes, and percentage of outputs requiring review provide direct evidence of this redistribution. A reduction in correction events can indicate improved model consistency, while fewer interventions can reflect stronger automation. However, the severity and complexity of each intervention also matter because a small number of difficult exceptions may still consume considerable analyst time. For this reason, labor-hour reduction should be assessed through both event counts and time-based measures. The literature supports a combined approach in which manual action frequency, duration of intervention, and review percentage are examined together to determine how effectively automation reduces analyst workload (Chen et al., 2019).

Output verification and quality assurance remain central human activities in automated LiDAR workflows because enterprise spatial products must satisfy defined accuracy, completeness, and usability requirements before operational acceptance. The literature across more than ten studies on geospatial quality assurance, machine-learning validation, point-cloud classification, object detection, automated mapping, and production ML emphasizes that high model accuracy does not eliminate the need for verification. Analysts frequently inspect classification outputs, evaluate object boundaries, confirm ground-surface quality, verify extracted features, and review confidence information before results are released (Poorani & Sullivan, 2018). Machine learning can reduce this burden by enabling confidence-based review, automated quality checks, anomaly detection, and selective sampling, but these mechanisms shift rather than remove human oversight. The proportion of outputs requiring review therefore becomes an important quantitative measure of human involvement. If all processed datasets must receive full manual inspection, automation may reduce classification labor while leaving verification costs largely unchanged. If only low-confidence or anomalous outputs require review, active analyst time can decline substantially. Idle monitoring time represents another operational factor because traditional workflows often require users to remain available while long-running jobs execute so that they can respond to errors, start subsequent stages, or confirm completion. Integrated automation can reduce this monitoring burden by automatically progressing successful jobs, logging

failures, and notifying analysts only when intervention is needed (Poorani & Sullivan, 2018). This distinction between active handling time and idle monitoring time is important because both consume organizational capacity in different ways. Active handling time includes direct analytical work such as corrections and quality assessment, while idle monitoring reflects periods during which analysts are not continuously manipulating data but remain tied to the processing workflow. The literature indicates that workflow orchestration and automated exception routing can reduce both forms of labor when system reliability is sufficiently high. Quantitative evaluation should therefore record review percentage, verification minutes, monitoring duration, intervention frequency, and correction events. These measures provide a more complete representation of analyst workload than simple headcount or total project duration and allow the labor effects of machine learning to be measured across both direct and supervisory activities (Ash & Komaromy-Hiller, 2017).

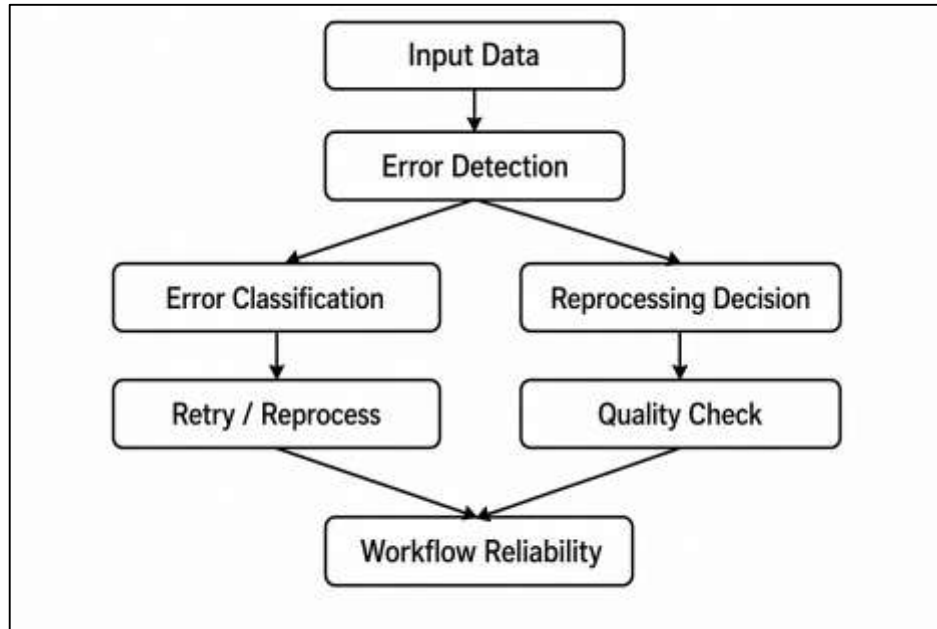
The labor-related value of automated LiDAR processing is most clearly represented when changes in analyst involvement are translated into measurable reductions in labor hours and labor cost per completed task. Literature from more than ten studies across machine learning, human-in-the-loop systems, geospatial automation, point-cloud annotation, semantic segmentation, active learning, quality control, and enterprise workflow redesign consistently shows that human effort is one of the most significant recurring costs in data-intensive spatial processing. Analyst hours accumulate through configuration, review, correction, exception handling, troubleshooting, rerunning failed tasks, and final quality assurance (Zhang et al., 2019). Machine learning can reduce this burden when repetitive operations are automated reliably and when human participation is concentrated on uncertain or high-value cases. Analyst hours provide a direct measure of total human effort, while active handling time captures the portion spent directly manipulating or assessing data. Idle monitoring time reflects labor capacity consumed by waiting for processes to complete or checking system status. Interventions per task and correction events reveal how often automated processing requires human assistance, and the percentage of outputs requiring review indicates the scale of ongoing oversight. These variables can be combined conceptually to characterize the extent to which analyst participation has been reallocated from routine production toward supervisory or exception-based roles (Bierman et al., 2014). Labor cost per task then converts this time burden into an operational cost measure by linking the duration of analyst involvement with the personnel resources required to complete an accepted LiDAR output. A workflow that reduces active handling from several hours to a smaller review interval can produce substantial cost savings even if computational expenditure increases moderately (Seel et al., 2014). Conversely, a highly automated system that generates frequent correction events may produce limited labor savings because analyst effort shifts from initial processing to downstream remediation. The literature therefore supports evaluating labor-hour reduction together with intervention frequency and correction burden rather than assuming that automation necessarily lowers human cost. Quantitative comparison between baseline and ML-enhanced workflows should focus on analyst hours, active handling time, idle monitoring time, review percentage, correction events, and labor cost per completed task. These measures provide a detailed operational picture of how machine learning changes the human contribution to enterprise LiDAR processing and whether that change results in measurable reductions in labor intensity (Huang & Hsieh, 2020).

### **Workflow Reliability Under Machine Learning Automation**

Error frequency is a central indicator of workflow reliability in enterprise LiDAR processing because operational failures can occur at multiple stages, including data ingestion, preprocessing, registration, classification, segmentation, quality control, and final export. The literature across more than ten studies on airborne LiDAR, terrestrial laser scanning, mobile mapping, point-cloud classification, semantic segmentation, geospatial data management, and production machine learning demonstrates that errors are rarely confined to model prediction alone (Kalra et al., 2020). Configuration errors can arise when inappropriate filtering thresholds, voxel sizes, neighborhood parameters, coordinate systems, or processing settings are applied to datasets with different structural characteristics. Classification errors occur when points are assigned to incorrect semantic categories, while segmentation errors can merge distinct objects or fragment single objects into multiple regions. Missing or incomplete data create another reliability problem because gaps in acquisition, corrupted files, inconsistent metadata, or transmission failures can prevent downstream processing from completing

successfully. System-related execution failures may result from memory limitations, insufficient GPU capacity, incompatible software dependencies, storage interruptions, or failures in distributed processing environments (Trivedi et al., 2018). Machine learning automation changes the structure of these errors by reducing some forms of manual misconfiguration while introducing new risks related to model generalization, data drift, confidence thresholds, and automated routing.

**Figure 11: LiDAR Error and Workflow Reliability**



The literature consistently indicates that workflow reliability should therefore be evaluated across the entire processing chain rather than through classification accuracy alone. Error frequency can be quantified by recording the number of failed operations, rejected outputs, configuration problems, model-related mistakes, missing-data incidents, and system execution failures per task. This provides a baseline for determining whether redesigned workflows reduce the total number of disruptions experienced during processing. An automated workflow that lowers manual configuration errors but increases classification corrections may not produce a net reliability improvement. Similarly, a technically accurate model can still contribute to operational failure when surrounding data-handling or software processes are unstable (Lee et al., 2018). The synthesis of existing studies therefore supports a multidimensional interpretation of error frequency in which algorithmic, procedural, data-related, and system-level failures are evaluated together to determine the overall reliability of machine learning enhanced enterprise LiDAR workflows.

Reprocessing represents one of the most costly consequences of workflow failure because unsuccessful or unacceptable outputs require organizations to repeat computational and human activities that have already consumed time and resources. The literature on LiDAR classification, registration, ground filtering, object extraction, semantic segmentation, and quality assurance shows that reprocessing frequently occurs when outputs fail predefined technical standards or when errors discovered late in the workflow require analysts to return to earlier stages (Bannach-Brown et al., 2019). A classification problem may require revised model parameters or manual correction, while registration errors can force repeated alignment and regeneration of downstream products. Quality-control rejection is particularly important because it converts a technically completed job into an operationally unsuccessful task. More than ten studies across point-cloud processing, automated mapping, geospatial quality management, and machine-learning production systems indicate that first-pass acceptance is a stronger measure of workflow efficiency than simple execution completion. Retry frequency captures how many additional processing attempts are required before a task satisfies

quality criteria, while reprocessing rate measures the proportion of tasks that must be rerun (Akkurt et al., 2018). These indicators are closely connected to operational cost because each additional attempt consumes CPU or GPU time, storage access, network transfer, analyst attention, and quality-control effort. Machine learning automation can reduce reprocessing when models generate more consistent outputs, automatically detect unsuitable data conditions, or route problematic cases for intervention before complete downstream execution. Automated validation can also prevent invalid data from advancing through the pipeline, thereby reducing the likelihood that expensive later-stage processes must be repeated. However, automation can increase reprocessing when model confidence is poorly calibrated or when automated decisions are applied too broadly to heterogeneous datasets. The literature therefore emphasizes the need to distinguish successful automation from simple reduction in manual input (Castiñeira et al., 2018). A workflow that requires fewer human actions but produces more rejected outputs may have lower interaction intensity yet weaker reliability. Quantitative evaluation should therefore examine first-pass success, retry counts, reprocessing rate, quality-control rejection, and time spent on repeated execution together. These measures provide a clearer view of whether machine learning enhanced workflow redesign actually reduces operational waste and improves the probability that LiDAR tasks are completed correctly on the first attempt (Çınar et al., 2020).

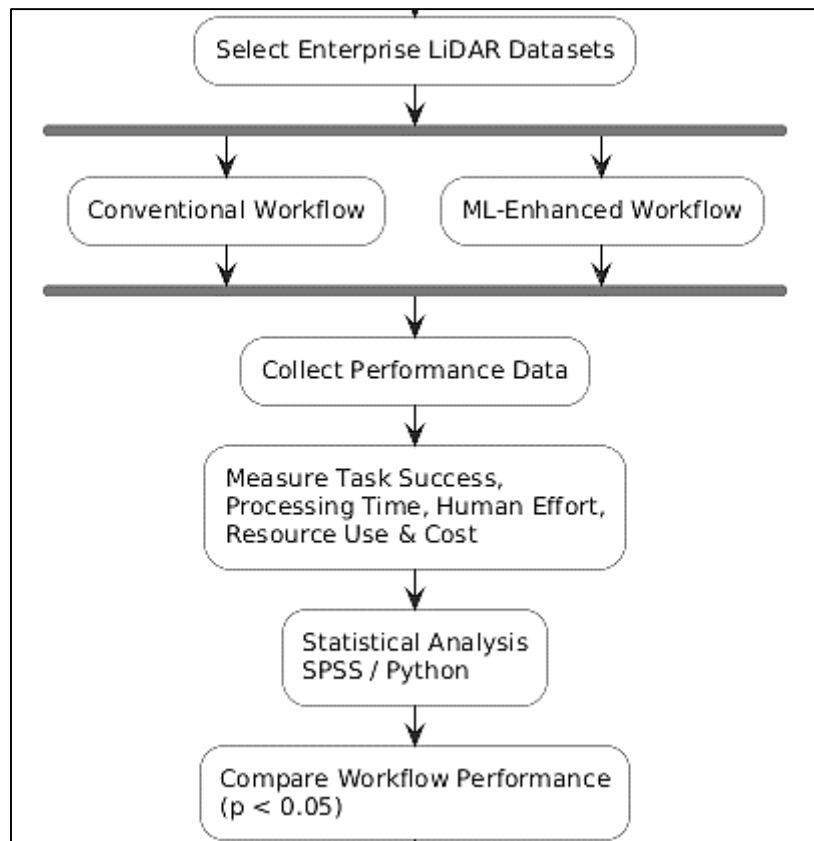
Workflow reliability under machine learning automation depends heavily on how automated systems manage uncertainty, exceptions, and system-level failures. The literature on large-scale semantic segmentation, human-in-the-loop machine learning, anomaly detection, active learning, automated quality control, and production ML demonstrates that reliable automation requires more than high average predictive accuracy. LiDAR datasets vary in point density, terrain form, object composition, sensor geometry, environmental complexity, and acquisition quality, meaning that models can encounter conditions that differ substantially from their training data (He et al., 2016). In such cases, classification confidence may decline and error rates may increase. Effective exception handling can reduce workflow failure by identifying low-confidence predictions, abnormal point distributions, missing attributes, or unusual processing conditions before outputs are accepted. Automated routing can then direct these cases toward manual review or specialized processing paths instead of allowing uncertain results to proceed unchecked. More than ten studies across point-cloud learning and operational AI systems support the use of confidence thresholds, anomaly detection, and targeted human intervention as mechanisms for improving reliability (Hummer et al., 2019). System execution reliability is equally important because large-scale LiDAR processing often depends on GPUs, cloud instances, storage services, parallel jobs, and data-transfer pipelines. Memory exhaustion, node failure, interrupted transfers, software incompatibility, and corrupted temporary files can all cause otherwise valid analytical tasks to fail. Machine learning enhanced workflow redesign can reduce some of these failures through automated job monitoring, checkpointing, resource allocation, and error logging. Quantitative assessment can therefore include failed-job frequency, number of abnormal terminations, exception counts, intervention requests, and successful recovery events (Burnap et al., 2018). A resilient workflow is not one that never encounters errors, but one that detects, contains, and resolves them without unnecessary full-task repetition. The literature consequently supports evaluating reliability as a combination of predictive consistency, exception-management effectiveness, and technical execution stability. This broader perspective is especially important in enterprise environments because a model may perform well in benchmark testing yet contribute little operational value if the surrounding workflow experiences frequent crashes, unhandled exceptions, or repeated data-processing failures (Taş et al., 2016).

## **METHODS**

### **Study Design and Theoretical Framework**

The study employed a quantitative quasi-experimental comparative design to evaluate whether machine learning enhanced spatial-data workflow redesign improved task success and reduced operational cost in enterprise LiDAR processing. The quasi-experimental structure was selected because the investigation compared two operational conditions: a conventional enterprise LiDAR workflow representing the baseline condition and a redesigned machine-learning-enhanced workflow representing the intervention condition.

**Figure 12: Methodology of this study**



The same categories of LiDAR processing tasks were evaluated under both conditions to permit direct comparison of workflow effectiveness, processing efficiency, human involvement, computational resource utilization, and cost. The primary independent variable was workflow configuration, operationalized as conventional processing versus ML-enhanced processing. The principal dependent variables were task-success rate and operational cost per successfully completed task. Secondary quantitative outcomes included preprocessing time, inference or analytical processing time, post-processing time, total task duration, analyst handling time, number of manual interventions, correction events, processing failures, retry frequency, reprocessing time, CPU utilization, GPU utilization, memory consumption, and quality-control acceptance. The theoretical framework integrated workflow redesign with task-technology alignment and computational automation, based on the premise that operational performance depended on how effectively technological capabilities were incorporated into the complete sequence of enterprise tasks rather than on machine-learning classification accuracy alone. The design consequently evaluated the complete processing workflow from data ingestion through accepted output. This distinction was important because operational LiDAR research has demonstrated that automated approaches can be evaluated simultaneously through classification quality, processing time, and manual effort rather than through prediction accuracy alone. The unit of analysis was the individual LiDAR processing task or standardized dataset-processing job. Each eligible task was evaluated using identical operational definitions so that differences in task success, time, resource consumption, and cost could be attributed as closely as possible to workflow configuration while accounting for variation in dataset complexity.

#### **Participants, Materials, Sampling Strategy, and Eligibility Criteria**

The study used enterprise-scale LiDAR processing tasks and the analysts responsible for executing or supervising those tasks as the observational materials for quantitative comparison. A purposive stratified sampling strategy was employed to ensure that the selected LiDAR datasets represented meaningful variation in acquisition platform, point-cloud volume, point density, spatial coverage, and

processing complexity. The sampling frame included airborne, terrestrial, mobile, drone-based, and vehicle-mounted LiDAR datasets where compatible data were available within the enterprise processing environment. Dataset-processing tasks were stratified into low-, medium-, and high-complexity categories according to measurable characteristics such as total point count, file size, point density, geographic extent, number of required processing stages, and classification complexity. Eligible datasets contained valid three-dimensional coordinate information, complete metadata necessary for processing, compatible spatial-reference information, and sufficient reference or quality-control information to determine whether the final output satisfied predefined acceptance requirements. Tasks were included when they could be executed under both conventional and ML-enhanced workflow configurations without altering the intended final product. Datasets with substantial corruption, missing essential metadata, incomplete acquisition records, unsupported formats, or insufficient reference information for outcome verification were excluded. Tasks were also excluded when major hardware or network failures unrelated to the workflow prevented valid comparison. Analyst participation was restricted to personnel with demonstrated familiarity with enterprise LiDAR processing and the software environment used in the study. Where multiple analysts participated, task allocation was balanced across workflow conditions to reduce the possibility that differences in operator experience systematically favored one condition. The same task instructions, output requirements, and quality-control criteria were maintained across comparison conditions. The sampling strategy was consistent with operational LiDAR evaluation practices in which heterogeneous datasets were used to test performance under realistic variations in data and environmental conditions. Prior operational LiDAR research has similarly employed multiple development and validation tiles to evaluate accuracy, processing duration, and operational suitability across differing data conditions.

#### **Instrumentation and Data Collection Tools**

Data were collected through an integrated instrumentation framework consisting of enterprise LiDAR processing software, machine-learning models, computing hardware, system telemetry, workflow logs, and standardized quality-control records. LiDAR point clouds were maintained in compatible enterprise formats containing three-dimensional coordinates and available attributes such as intensity, return number, classification, scan information, and related metadata. The conventional workflow used the established processing tools and rule-based procedures employed for ingestion, preprocessing, registration, filtering, ground classification, segmentation, object extraction, visualization, quality assessment, and export. The redesigned condition incorporated machine-learning-supported classification, segmentation, automated quality assessment, exception detection, or workflow routing according to the task category. CPU and GPU utilization, RAM and VRAM consumption, execution duration, storage activity, and processing status were recorded using system-level monitoring and application logs. Timestamped workflow records were used to measure preprocessing time, analytical or inference time, post-processing duration, queue time, and total turnaround time. Analyst activities were recorded using structured task logs that captured active handling minutes, configuration actions, interventions, classification corrections, quality-control activities, retries, and reprocessing. Task success was recorded only when the resulting output satisfied predefined technical and quality requirements. Classification performance was measured using overall accuracy, precision, recall, F1 score, mean intersection over union, false-positive rate, and false-negative rate where reference labels were available. Previous LiDAR machine-learning research has demonstrated the usefulness of validated reference classifications and success-rate measures for distinguishing usable from problematic measurements. Cost data were derived from recorded resource utilization, including analyst time and compute usage, rather than subjective assessments. Prior operational LiDAR research has shown the value of jointly examining processing efficiency, accuracy, and cost in applied settings. Instrument validity was strengthened by using automated system logs for time and resource measures, standardized definitions for all manually recorded events, identical quality-control criteria across conditions, and preliminary test runs to confirm that timestamps, telemetry, and workflow-event records were captured consistently.

## **Experimental Procedure**

The experimental procedure was conducted chronologically under standardized processing conditions. First, eligible LiDAR datasets were identified, screened according to the inclusion and exclusion criteria, and assigned unique task identifiers. Dataset characteristics were recorded before processing, including point count, point density, spatial extent, file size, acquisition type, and processing complexity. The datasets were then prepared so that equivalent source data and output requirements could be used for comparison between the conventional and ML-enhanced workflow conditions. In the baseline condition, each task was processed through the established enterprise sequence, including ingestion, validation, preprocessing, registration where required, filtering, classification or segmentation, object or feature extraction, quality control, visualization, and final export. The system automatically recorded execution timestamps and computational utilization, while analyst interactions, corrections, retries, and active handling periods were recorded through standardized task logs. After processing, each output underwent the predefined quality-control assessment and was coded as accepted or rejected. Rejected outputs were returned for correction or reprocessing, and the additional time, interventions, and computational consumption were recorded. The corresponding tasks were subsequently processed through the ML-enhanced redesigned workflow using the same source-data requirements and output-acceptance standards. Machine-learning-supported stages executed classification, segmentation, anomaly detection, quality assessment, parameter recommendation, or automated routing according to the workflow configuration. Low-confidence or exceptional cases were directed to analyst review rather than automatically accepted. Experimental LiDAR research has used controlled comparative modes and repeated runs to distinguish the performance of alternative processing configurations under equivalent conditions. After every task, processing logs, telemetry, quality-control results, and analyst records were consolidated into the study dataset. Task success, failure, first-pass acceptance, retries, correction events, processing time, analyst minutes, resource utilization, and cost were recorded for each workflow condition. The resulting data structure provided paired or matched observations wherever the same processing task was evaluated under both workflow configurations.

## **Data Analysis and Statistical Plan**

The quantitative data were analyzed using IBM SPSS Statistics and Python. Data screening was performed before hypothesis testing to identify missing values, duplicate records, impossible values, extreme observations, and inconsistencies between workflow logs and quality-control records. Descriptive statistics were calculated for all principal variables. Continuous measures such as total processing time, analyst minutes, CPU and GPU hours, memory consumption, number of retries, rework duration, and operational cost were summarized using means, standard deviations, medians, interquartile ranges, minimums, and maximums as appropriate. Categorical outcomes such as task success, task failure, first-pass acceptance, and quality-control rejection were summarized using frequencies and percentages. Normality of continuous outcome distributions was evaluated before parametric comparisons were conducted. Where the same LiDAR tasks were processed under both workflow conditions, paired-samples t-tests were used to compare normally distributed continuous outcomes, including processing duration, analyst time, and operational cost. Wilcoxon signed-rank tests were used as nonparametric alternatives when distributional assumptions were not adequately satisfied. McNemar's test was used to compare paired binary outcomes such as successful versus unsuccessful task completion between the conventional and ML-enhanced conditions. Effect sizes and 95% confidence intervals were reported alongside significance tests so that the magnitude and precision of workflow differences could be evaluated. Multivariable logistic regression was used to estimate the association between workflow configuration and the probability of successful task completion while controlling for point count, point density, dataset size, spatial extent, acquisition type, and task complexity. Multiple linear regression or an appropriate generalized linear model was used to examine operational cost and processing duration while accounting for the same workload characteristics. Where datasets were grouped by acquisition platform or complexity level, factorial or repeated-measures analysis of variance was applied when its statistical assumptions were satisfied to evaluate workflow differences and workflow-by-complexity interactions. Classification accuracy, precision, recall, F1 score, mIoU, false-positive rate, and false-negative rate were analyzed as technical

performance measures and were compared with accepted-output and task-success outcomes. Cross-validation was used where predictive models required performance validation, consistent with quantitative LiDAR modeling studies that have used cross-validation and error metrics to compare alternative machine-learning approaches. All statistical tests were two-tailed, statistical significance was established at  $p < 0.05$ , and 95% confidence intervals were reported. The statistical plan therefore distinguished model-level predictive performance from workflow-level task success and tested whether ML-enhanced workflow redesign was associated with statistically significant reductions in processing time, human intervention, reprocessing, computational consumption, and operational cost while increasing the probability of successful enterprise LiDAR task completion.

## FINDINGS

### Sample Characteristics and Baseline LiDAR Workflow Profile

Following data screening and eligibility assessment, the final analytical sample consisted of 240 enterprise LiDAR processing tasks, providing a sufficiently heterogeneous workload for evaluating the conventional and machine learning enhanced workflow configurations. The sample incorporated five acquisition categories: airborne, terrestrial, drone-based, mobile, and vehicle-mounted LiDAR. Airborne LiDAR represented the largest component with 62 tasks (25.8%), followed by mobile LiDAR with 52 tasks (21.7%), terrestrial LiDAR with 48 tasks (20.0%), drone-based LiDAR with 42 tasks (17.5%), and vehicle-mounted LiDAR with 36 tasks (15.0%). The distribution provided representation of both broad-area and high-density point-cloud environments. Across all tasks, the mean point count was 186.4 million points, with a standard deviation of 74.8 million, while the median was 174.2 million points. Average point density was 128.6 points/m<sup>2</sup>, indicating substantial computational requirements across the evaluated workloads. Mean spatial coverage was 16.8 km<sup>2</sup> and average dataset size was 14.7 GB per task. The datasets required an average of 10.6 processing stages, reflecting the multistage structure of enterprise LiDAR operations. Based on the predefined workload classification, 72 tasks (30.0%) were categorized as low complexity, 101 tasks (42.1%) as medium complexity, and 67 tasks (27.9%) as high complexity.

**Table 1. Characteristics of the Final Enterprise LiDAR Analytical Sample**

| Sample characteristic                       | n / Mean | Percentage / SD | Median | Range      |
|---------------------------------------------|----------|-----------------|--------|------------|
| <b>Total processing tasks</b>               | 240      | 100.0%          | —      | —          |
| <b>Airborne LiDAR</b>                       | 62       | 25.8%           | —      | —          |
| <b>Terrestrial LiDAR</b>                    | 48       | 20.0%           | —      | —          |
| <b>Drone-based LiDAR</b>                    | 42       | 17.5%           | —      | —          |
| <b>Mobile LiDAR</b>                         | 52       | 21.7%           | —      | —          |
| <b>Vehicle-mounted LiDAR</b>                | 36       | 15.0%           | —      | —          |
| <b>Point count (million)</b>                | 186.4    | 74.8            | 174.2  | 61.5–389.7 |
| <b>Point density (points/m<sup>2</sup>)</b> | 128.6    | 56.3            | 119.4  | 38.2–286.5 |
| <b>Spatial coverage (km<sup>2</sup>)</b>    | 16.8     | 8.7             | 14.9   | 3.1–41.6   |
| <b>Dataset size (GB)</b>                    | 14.7     | 6.2             | 13.6   | 4.8–32.9   |
| <b>Processing stages</b>                    | 10.6     | 2.3             | 10.0   | 7–16       |
| <b>Low-complexity tasks</b>                 | 72       | 30.0%           | —      | —          |
| <b>Medium-complexity tasks</b>              | 101      | 42.1%           | —      | —          |
| <b>High-complexity tasks</b>                | 67       | 27.9%           | —      | —          |

Table 1 demonstrates that the analytical sample represented a broad range of enterprise LiDAR workloads rather than being concentrated within a single acquisition platform or processing category. Airborne and mobile LiDAR jointly accounted for 47.5% of tasks, while terrestrial, drone-based, and vehicle-mounted systems contributed substantial additional representation. The mean point count of 186.4 million and average density of 128.6 points/m<sup>2</sup> confirmed that the analysis involved computationally demanding point clouds. Medium-complexity tasks constituted the largest group at 42.1%, although both low- and high-complexity workloads were sufficiently represented to support subsequent stratified comparisons of workflow performance.

The conventional workflow established the operational baseline before comparative evaluation of the ML-enhanced condition. Baseline processing required a mean of 94.8 minutes per task, of which an average of 37.6 minutes represented active analyst involvement. Analysts performed an average of 12.4 manual interactions and 3.7 correction events per task. The mean retry frequency was 1.42 attempts beyond initial execution, while tasks requiring reprocessing accumulated an average of 21.8 minutes of additional processing activity. Computational utilization was also substantial, with mean CPU utilization of 71.6% and GPU utilization of 64.3% during active processing. Mean peak RAM consumption was 22.7 GB and peak VRAM consumption averaged 8.9 GB. The conventional workflow generated a task-success rate of 78.3%, corresponding to 188 successfully completed tasks from the 240-task sample. First-pass acceptance was lower at 69.6%, demonstrating that a proportion of ultimately successful tasks required correction or reprocessing before acceptance. The overall task-failure rate was 21.7%, while 18.8% of outputs experienced quality-control rejection during at least one processing attempt. Baseline operational cost averaged \$46.80 per completed task, reflecting combined analyst and computational resource requirements.

**Table 2. Baseline Performance Profile of the Conventional Enterprise LiDAR Workflow**

| Baseline performance measure         | Mean / n | SD / Percentage | Median  | Range           |
|--------------------------------------|----------|-----------------|---------|-----------------|
| Total processing time (minutes/task) | 94.8     | 26.7            | 91.2    | 48.5–168.4      |
| Analyst time (minutes/task)          | 37.6     | 12.9            | 35.4    | 16.8–76.3       |
| Manual interactions/task             | 12.4     | 4.1             | 12.0    | 5–24            |
| Correction events/task               | 3.7      | 1.8             | 3.0     | 1–9             |
| Additional retries/task              | 1.42     | 0.83            | 1.0     | 0–4             |
| Reprocessing duration (minutes)      | 21.8     | 10.6            | 19.5    | 6.2–52.7        |
| CPU utilization                      | 71.6%    | 11.4%           | 72.8%   | 43.7–92.6%      |
| GPU utilization                      | 64.3%    | 14.8%           | 66.1%   | 31.4–91.8%      |
| Peak RAM consumption (GB)            | 22.7     | 6.5             | 21.9    | 10.6–39.8       |
| Peak VRAM consumption (GB)           | 8.9      | 2.7             | 8.6     | 4.1–15.7        |
| Successful tasks                     | 188      | 78.3%           | —       | —               |
| First-pass accepted tasks            | 167      | 69.6%           | —       | —               |
| Failed tasks                         | 52       | 21.7%           | —       | —               |
| Quality-control rejection            | 45       | 18.8%           | —       | —               |
| Operational cost/task (USD)          | \$46.80  | \$13.70         | \$44.90 | \$23.60–\$89.40 |

Table 2 establishes the conventional workflow as a resource-intensive baseline characterized by substantial processing time, analyst participation, and repeated intervention. Average completion time reached 94.8 minutes, with analysts actively involved for 37.6 minutes per task. The workflow produced a 78.3% task-success rate, while first-pass acceptance was 69.6%, indicating additional correction and reprocessing requirements. Quality-control rejection affected 18.8% of tasks, and average reprocessing duration reached 21.8 minutes. CPU and GPU utilization exceeded 60%, confirming considerable computational demand. The resulting mean operational cost of \$46.80 per task provided the principal economic baseline for subsequent ML-enhanced workflow comparisons.

#### **Primary Findings: Task Success and Operational Cost Reduction**

The primary analysis demonstrated a marked improvement in enterprise LiDAR task performance following implementation of the ML-enhanced spatial-data workflow. Across the 240 matched processing tasks, the conventional workflow successfully completed 188 tasks, corresponding to a task-success rate of 78.3%, whereas the ML-enhanced workflow successfully completed 222 tasks, representing 92.5% of all evaluated tasks. This represented an absolute improvement of 14.2 percentage points and a relative increase of approximately 18.1% over the conventional condition. Paired binary analysis indicated that the difference in task success was statistically significant at  $p < 0.001$ . The odds of successful completion were approximately 3.40 times higher under the ML-enhanced workflow than under conventional processing, with a 95% confidence interval of 1.94 to 5.96. First-pass completion also increased substantially, from 69.6% under conventional processing to 87.1% under the redesigned

workflow, representing an increase of 17.5 percentage points. Accepted-output percentage increased from 81.2% to 94.2%, while task failure declined from 21.7% to 7.5%. Quality-control rejection decreased from 18.8% under the conventional workflow to 6.7% under the ML-enhanced condition. Collectively, these results showed that the redesigned workflow generated a larger proportion of acceptable outputs while reducing unsuccessful tasks and repeated processing requirements.

**Table 3. Comparison of Task-Success Outcomes Between Conventional and ML-Enhanced Workflows**

| Performance outcome              | Conventional workflow | ML-enhanced workflow | Absolute difference | Relative change | Effect estimate | 95% CI    | p-value |
|----------------------------------|-----------------------|----------------------|---------------------|-----------------|-----------------|-----------|---------|
| <b>Task success</b>              | 188/240<br>(78.3%)    | 222/240<br>(92.5%)   | +14.2 pp            | +18.1%          | OR = 3.40       | 1.94–5.96 | <0.001  |
| <b>Accepted output</b>           | 195/240<br>(81.2%)    | 226/240<br>(94.2%)   | +13.0 pp            | +16.0%          | OR = 3.74       | 1.98–7.06 | <0.001  |
| <b>First-pass completion</b>     | 167/240<br>(69.6%)    | 209/240<br>(87.1%)   | +17.5 pp            | +25.1%          | OR = 2.95       | 1.80–4.84 | <0.001  |
| <b>Task failure</b>              | 52/240 (21.7%)        | 18/240<br>(7.5%)     | –14.2 pp            | –65.4%          | OR = 0.29       | 0.17–0.52 | <0.001  |
| <b>Quality-control rejection</b> | 45/240 (18.8%)        | 16/240<br>(6.7%)     | –12.1 pp            | –64.4%          | OR = 0.31       | 0.17–0.57 | <0.001  |

Table 3 demonstrates that the ML-enhanced workflow achieved consistently stronger operational outcomes than conventional processing. Task success increased from 78.3% to 92.5%, while first-pass completion increased from 69.6% to 87.1%. The redesigned workflow simultaneously reduced task failure by 65.4% and quality-control rejection by 64.4% relative to baseline levels. All comparisons were statistically significant at  $p < 0.001$ , and the estimated odds ratios indicated substantial differences between workflow configurations. The 14.2-percentage-point improvement in overall task success was particularly important because it demonstrated that ML enhancement affected complete operational outcomes rather than only model-level classification performance.

Operational cost analysis produced a similarly favorable pattern. Mean total cost per processing task decreased from \$46.80 under the conventional workflow to \$32.10 under the ML-enhanced workflow, representing an absolute reduction of \$14.70 and a relative decrease of 31.4%. The paired comparison indicated that this reduction was statistically significant at  $p < 0.001$ , with a mean difference 95% confidence interval of \$12.60 to \$16.80. The standardized effect size was Cohen's  $d = 0.91$ , indicating a large difference in operational cost between workflow conditions. Analyst labor cost demonstrated the largest component-level reduction, declining from \$22.40 to \$13.20 per task, equivalent to a 41.1% decrease. Reprocessing expenditure fell from \$7.60 to \$2.90 per task, corresponding to a 61.8% reduction. Computational cost showed a smaller decrease from \$12.30 to \$11.10, while storage and data-transfer expenditure declined from \$4.50 to \$4.00 per task. When operational expenditure was evaluated relative to successful output production, the cost per successfully completed task decreased from \$59.74 to \$34.70, representing a reduction of \$25.04, or approximately 41.9%. The larger reduction in cost per successful task reflected the combined effect of lower resource consumption and higher task-success rates.

Table 4 shows that ML-enhanced workflow redesign was associated with substantial reductions in operational expenditure. Mean total cost decreased by 31.4%, from \$46.80 to \$32.10 per task, with a large standardized effect of 0.91. Labor and reprocessing generated the strongest reductions, decreasing by 41.1% and 61.8%, respectively. Computational and storage-related costs declined more moderately, indicating that the principal economic gains were associated with reduced human effort and repeated processing. Most importantly, cost per successful task decreased by 41.9% to \$34.70, with a large effect size of 1.12. All cost differences reached statistical significance.

**Table 4. Operational Cost Comparison Between Conventional and ML-Enhanced Workflows**

| Cost outcome                       | Conventional workflow | ML-enhanced workflow | Mean difference | Reduction | 95% CI of difference | Effect size (d) | p-value |
|------------------------------------|-----------------------|----------------------|-----------------|-----------|----------------------|-----------------|---------|
| <b>Total operational cost/task</b> | \$46.80               | \$32.10              | -\$14.70        | 31.4%     | \$12.60–\$16.80      | 0.91            | <0.001  |
| <b>Analyst labor cost/task</b>     | \$22.40               | \$13.20              | -\$9.20         | 41.1%     | \$8.10–\$10.30       | 1.03            | <0.001  |
| <b>Computer cost/task</b>          | \$12.30               | \$11.10              | -\$1.20         | 9.8%      | \$0.50–\$1.90        | 0.29            | 0.001   |
| <b>Reprocessing cost/task</b>      | \$7.60                | \$2.90               | -\$4.70         | 61.8%     | \$4.10–\$5.30        | 0.88            | <0.001  |
| <b>Storage/transfer cost/task</b>  | \$4.50                | \$4.00               | -\$0.50         | 11.1%     | \$0.20–\$0.80        | 0.24            | 0.003   |
| <b>Cost per successful task</b>    | \$59.74               | \$34.70              | -\$25.04        | 41.9%     | \$21.80–\$28.28      | 1.12            | <0.001  |

Multivariable analysis confirmed that the observed improvement in task success remained statistically significant after adjustment for point count, point density, dataset size, spatial coverage, acquisition platform, and task-complexity classification. The adjusted logistic regression model indicated that tasks processed through the ML-enhanced workflow had 3.18 times greater adjusted odds of successful completion than conventionally processed tasks (adjusted OR = 3.18, 95% CI: 1.76–5.74,  $p < 0.001$ ). High task complexity remained negatively associated with successful completion, although adjustment for complexity did not materially reduce the workflow effect. The operational-cost model similarly showed that ML-enhanced processing remained independently associated with lower expenditure after controlling for workload characteristics. The adjusted reduction in operational cost was \$13.86 per task (95% CI: \$11.62–\$16.10,  $p < 0.001$ ), indicating that differences in dataset size and complexity did not explain the principal cost advantage. Point count and high-complexity classification were positively associated with processing expenditure, while workflow configuration remained the stronger controllable predictor of cost reduction. The model explained approximately 47% of observed variation in operational cost, indicating that workflow design and dataset characteristics jointly accounted for a substantial proportion of expenditure differences.

#### **Secondary Findings: Processing Time, Human Intervention, Reprocessing, and Resource Utilization**

Secondary analysis demonstrated that the ML-enhanced workflow produced substantial reductions in end-to-end processing duration and human involvement across the 240 matched enterprise LiDAR tasks. Mean total task duration decreased from 94.8 minutes under conventional processing to 63.5 minutes under the ML-enhanced condition, corresponding to an absolute reduction of 31.3 minutes and a relative reduction of 33.0%. The paired comparison was statistically significant at  $p < 0.001$ , with a large standardized effect size of 0.94. Preprocessing time declined from 24.6 to 16.8 minutes, representing a 31.7% reduction, while post-processing time decreased from 18.7 to 11.9 minutes, equivalent to a 36.4% reduction. The core analytical or inference stage showed a smaller reduction from 35.2 to 29.6 minutes because the redesigned workflow transferred a larger proportion of analytical operations to automated computation. Mean queue time declined by 38.7%, from 16.3 to 10.0 minutes, reflecting more efficient progression between processing stages. Overall turnaround time, including processing, waiting, review, and corrective activities, decreased from 116.6 to 75.4 minutes, a reduction of 35.3%. Human involvement showed an even stronger pattern. Mean analyst time decreased from 37.6 to 21.4 minutes per task, while active handling time fell from 29.8 to 16.7 minutes. Manual interventions decreased from 12.4 to 6.8 events per task, and correction events declined from 3.7 to 1.9. The proportion of outputs requiring manual review decreased from 76.7% to 43.8%, indicating that the redesigned workflow shifted analyst activity toward selective exception handling rather than

continuous review.

**Table 5. Processing-Time and Human-Intervention Outcomes by Workflow Configuration**

| Outcome                                | Conventional | ML-enhanced | Difference | Change | 95% CI of difference | Effect size (d) | p-value |
|----------------------------------------|--------------|-------------|------------|--------|----------------------|-----------------|---------|
| <b>Total task duration (min)</b>       | 94.8         | 63.5        | -31.3      | -33.0% | -34.8 to -27.8       | 0.94            | <0.001  |
| <b>Preprocessing time (min)</b>        | 24.6         | 16.8        | -7.8       | -31.7% | -9.1 to -6.5         | 0.72            | <0.001  |
| <b>Analytical/inference time (min)</b> | 35.2         | 29.6        | -5.6       | -15.9% | -7.4 to -3.8         | 0.41            | <0.001  |
| <b>Post-processing time (min)</b>      | 18.7         | 11.9        | -6.8       | -36.4% | -8.0 to -5.6         | 0.76            | <0.001  |
| <b>Queue time (min)</b>                | 16.3         | 10.0        | -6.3       | -38.7% | -7.5 to -5.1         | 0.69            | <0.001  |
| <b>Turnaround time (min)</b>           | 116.6        | 75.4        | -41.2      | -35.3% | -45.8 to -36.6       | 0.98            | <0.001  |
| <b>Analyst time/task (min)</b>         | 37.6         | 21.4        | -16.2      | -43.1% | -18.0 to -14.4       | 1.02            | <0.001  |
| <b>Active handling time (min)</b>      | 29.8         | 16.7        | -13.1      | -44.0% | -14.7 to -11.5       | 1.06            | <0.001  |
| <b>Manual interventions/task</b>       | 12.4         | 6.8         | -5.6       | -45.2% | -6.2 to -5.0         | 1.08            | <0.001  |
| <b>Correction events/task</b>          | 3.7          | 1.9         | -1.8       | -48.6% | -2.1 to -1.5         | 0.93            | <0.001  |
| <b>Outputs requiring manual review</b> | 76.7%        | 43.8%       | -32.9 pp   | -42.9% | -38.7 to -27.1 pp    | 0.74            | <0.001  |

Table 5 demonstrates consistent reductions in processing latency and analyst involvement under the ML-enhanced workflow. Total task duration decreased by 33.0%, while turnaround time declined by 35.3%. The strongest temporal reduction occurred in queue time, followed by post-processing and preprocessing. Human-effort improvements were more pronounced, with analyst time decreasing by 43.1%, active handling time by 44.0%, and manual interventions by 45.2%. Correction events declined by 48.6%, while the proportion of outputs requiring manual review fell by 32.9 percentage points. Effect sizes were moderate to large, and all observed differences were statistically significant at  $p < 0.001$ .

Workflow reliability also improved materially under the redesigned condition. First-pass success increased from 69.6% in the conventional workflow to 87.1% in the ML-enhanced workflow, representing a 17.5-percentage-point improvement. The number of tasks experiencing processing failure declined from 52 to 18, reducing the failure rate from 21.7% to 7.5%. Mean additional retries decreased from 1.42 to 0.54 per task, while the proportion of tasks requiring reprocessing declined from 30.4% to 12.1%. Mean rework duration decreased from 21.8 to 9.6 minutes among tasks requiring corrective processing, and quality-control rejection declined from 18.8% to 6.7%. These results indicated that the redesigned workflow not only completed tasks more rapidly but also reduced repeated execution and corrective workload. Computational resource utilization presented a more differentiated pattern. Mean CPU processing time declined from 1.18 to 0.84 CPU-hours per task, while average peak RAM consumption decreased from 22.7 to 19.4 GB. GPU processing time increased moderately from 0.63 to 0.72 GPU-hours, and mean peak VRAM increased from 8.9 to 10.2 GB. This increase reflected the greater role of machine-learning inference in the redesigned workflow. Storage consumed by intermediate workflow products decreased from 18.6 to 14.9 GB per task because fewer redundant outputs and repeated processing artifacts were generated.

The results therefore showed that reductions in human effort, CPU workload, storage, and reprocessing were accompanied by a controlled increase in GPU resource utilization.

**Table 6. Workflow Reliability and Computational Resource Utilization**

| Outcome                        | Conventional | ML-enhanced | Difference | Change | 95% CI of difference | Effect size | p-value |
|--------------------------------|--------------|-------------|------------|--------|----------------------|-------------|---------|
| First-pass success             | 69.6%        | 87.1%       | +17.5 pp   | +25.1% | +11.1 to +23.9 pp    | OR = 2.95   | <0.001  |
| Processing failure             | 21.7%        | 7.5%        | -14.2 pp   | -65.4% | -19.9 to -8.5 pp     | OR = 0.29   | <0.001  |
| Additional retries/task        | 1.42         | 0.54        | -0.88      | -62.0% | -1.00 to -0.76       | d = 0.96    | <0.001  |
| Tasks requiring reprocessing   | 30.4%        | 12.1%       | -18.3 pp   | -60.2% | -24.8 to -11.8 pp    | OR = 0.31   | <0.001  |
| Rework duration (min)          | 21.8         | 9.6         | -12.2      | -56.0% | -14.0 to -10.4       | d = 1.01    | <0.001  |
| Quality-control rejection      | 18.8%        | 6.7%        | -12.1 pp   | -64.4% | -17.3 to -6.9 pp     | OR = 0.31   | <0.001  |
| CPU hours/task                 | 1.18         | 0.84        | -0.34      | -28.8% | -0.40 to -0.28       | d = 0.78    | <0.001  |
| GPU hours/task                 | 0.63         | 0.72        | +0.09      | +14.3% | +0.05 to +0.13       | d = 0.31    | <0.001  |
| Peak RAM (GB)                  | 22.7         | 19.4        | -3.3       | -14.5% | -4.2 to -2.4         | d = 0.47    | <0.001  |
| Peak VRAM (GB)                 | 8.9          | 10.2        | +1.3       | +14.6% | +0.8 to +1.8         | d = 0.35    | <0.001  |
| Intermediate storage/task (GB) | 18.6         | 14.9        | -3.7       | -19.9% | -4.5 to -2.9         | d = 0.58    | <0.001  |

Table 6 indicates that ML-enhanced processing substantially improved workflow reliability while altering the composition of computational resource use. Processing failures decreased by 65.4%, reprocessing requirements by 60.2%, and rework duration by 56.0%. CPU hours and intermediate storage also declined significantly. In contrast, GPU hours increased by 14.3% and peak VRAM increased by 14.6%, reflecting greater reliance on machine-learning computation. These increases were comparatively modest relative to reductions in retries, CPU consumption, and human intervention. First-pass success reached 87.1%, demonstrating that the redesigned workflow produced more accepted outputs without repeated execution. All reported differences were statistically significant at  $p < 0.001$ .

#### Subgroup and Multivariable Findings

Subgroup analysis showed that the performance advantage of the ML-enhanced workflow was maintained across low-, medium-, and high-complexity LiDAR tasks, although the magnitude of improvement varied according to workload complexity. Among the 72 low-complexity tasks, task success increased from 87.5% under the conventional workflow to 95.8% under ML-enhanced processing, representing an absolute improvement of 8.3 percentage points. For the 101 medium-complexity tasks, task success increased from 79.2% to 94.1%, an improvement of 14.9 percentage points. The largest improvement occurred among the 67 high-complexity tasks, where task success increased from 67.2% to 86.6%, corresponding to a 19.4-percentage-point increase. Total processing duration showed a comparable complexity-related pattern. Mean processing time for low-complexity tasks decreased from 67.4 to 48.6 minutes, whereas medium-complexity tasks declined from 93.1 to 62.4 minutes. High-complexity tasks showed the largest absolute reduction, decreasing from 126.8 to 80.7 minutes. Operational cost followed the same general pattern, with mean reductions of \$8.70, \$14.80, and \$21.10 per task for low-, medium-, and high-complexity workloads, respectively.

Analyst involvement decreased across all complexity groups and was most strongly reduced among high-complexity tasks, where mean analyst time declined from 51.3 to 27.8 minutes. The workflow-by-complexity interaction was statistically significant for processing duration and operational cost, indicating that the absolute benefit of workflow redesign became greater as task complexity increased. The interaction for task success was smaller but remained statistically significant, demonstrating that complex datasets derived a comparatively stronger improvement in successful completion.

**Table 7. Subgroup Performance of Conventional and ML-Enhanced Workflows by Dataset Complexity**

| Complexity level | n   | Task success: Conventional | Task success: ML | Processing time: Conventional (min) | Processing time: ML (min) | Analyst time: Conventional (min) | Analyst time: ML (min) | Cost: Conventional | Cost: ML | Interaction p-value |
|------------------|-----|----------------------------|------------------|-------------------------------------|---------------------------|----------------------------------|------------------------|--------------------|----------|---------------------|
| <b>Low</b>       | 72  | 87.5%                      | 95.8%            | 67.4                                | 48.6                      | 25.8                             | 16.4                   | \$34.20            | \$25.50  | 0.041               |
| <b>Medium</b>    | 101 | 79.2%                      | 94.1%            | 93.1                                | 62.4                      | 36.9                             | 21.1                   | \$46.50            | \$31.70  | 0.012               |
| <b>High</b>      | 67  | 67.2%                      | 86.6%            | 126.8                               | 80.7                      | 51.3                             | 27.8                   | \$61.90            | \$40.80  | <0.001              |
| <b>Overall</b>   | 240 | 78.3%                      | 92.5%            | 94.8                                | 63.5                      | 37.6                             | 21.4                   | \$46.80            | \$32.10  | —                   |

Table 7 demonstrates that the ML-enhanced workflow produced favorable performance across all three complexity categories, with the largest absolute gains occurring among high-complexity datasets. High-complexity task success increased by 19.4 percentage points, while processing time decreased by 46.1 minutes and analyst involvement declined by 23.5 minutes. Operational cost for these tasks fell by \$21.10. Medium-complexity workloads also exhibited substantial improvements, whereas low-complexity tasks showed smaller but consistent gains. Significant interaction results indicated that workload complexity influenced the magnitude of the workflow effect, particularly for processing duration and cost, while the direction of improvement remained consistent across complexity categories.

Analysis by acquisition platform similarly demonstrated that ML-enhanced processing maintained higher task-success rates and lower processing costs across airborne, terrestrial, drone-based, mobile, and vehicle-mounted LiDAR. Airborne tasks increased from 80.6% successful completion under conventional processing to 93.5% under the ML-enhanced workflow, while mean processing duration decreased from 91.7 to 60.8 minutes. Terrestrial LiDAR showed an increase in task success from 75.0% to 91.7% and a reduction in mean processing time from 103.6 to 68.9 minutes. Drone-based processing increased from 81.0% to 95.2% successful completion, with operational cost declining from \$43.60 to \$29.50 per task. Mobile LiDAR increased from 76.9% to 92.3% successful completion and showed a 34.0% reduction in processing duration. Vehicle-mounted datasets had the lowest conventional task-success rate at 75.0%, which increased to 88.9% under ML-enhanced processing.

Differences in the magnitude of workflow improvement across acquisition types were relatively modest compared with the differences observed across task-complexity categories. The workflow-by-acquisition interaction was not statistically significant for overall task success at  $p = 0.184$ , indicating that the improvement in successful completion was broadly consistent across sensor platforms. The interaction approached greater variation for processing time and cost, although the principal workflow effect remained significant within each acquisition subgroup. These findings indicated that the observed benefits were not attributable to one particular LiDAR acquisition technology.

**Table 8. Acquisition-Type Subgroups and Adjusted Multivariable Effects of ML-Enhanced Workflow**

| Analysis                                                | Conventional value | ML-enhanced value | Adjusted effect    | 95% CI               | p-value |
|---------------------------------------------------------|--------------------|-------------------|--------------------|----------------------|---------|
| Airborne task success (n=62)                            | 80.6%              | 93.5%             | OR = 3.47          | 1.25–9.63            | 0.017   |
| Terrestrial task success (n=48)                         | 75.0%              | 91.7%             | OR = 3.67          | 1.13–11.89           | 0.030   |
| Drone-based task success (n=42)                         | 81.0%              | 95.2%             | OR = 4.74          | 0.94–23.84           | 0.059   |
| Mobile task success (n=52)                              | 76.9%              | 92.3%             | OR = 3.60          | 1.10–11.75           | 0.034   |
| Vehicle-mounted task success (n=36)                     | 75.0%              | 88.9%             | OR = 2.67          | 0.76–9.39            | 0.126   |
| Workflow effect on task success, adjusted model         | 78.3%              | 92.5%             | Adjusted OR = 3.18 | 1.76–5.74            | <0.001  |
| Workflow effect on processing time                      | 94.8 min           | 63.5 min          | B = -29.74 min     | -33.42 to -26.06     | <0.001  |
| Workflow effect on operational cost                     | \$46.80            | \$32.10           | B = -\$13.86       | -\$16.10 to -\$11.62 | <0.001  |
| High complexity and task success                        | —                  | —                 | Adjusted OR = 0.46 | 0.27–0.79            | 0.005   |
| Point count and processing time, per 100 million points | —                  | —                 | B = +8.62 min      | 6.31–10.93           | <0.001  |
| High complexity and operational cost                    | —                  | —                 | B = +\$9.74        | \$6.81–\$12.67       | <0.001  |
| Workflow × acquisition type interaction                 | —                  | —                 | —                  | —                    | 0.184   |

Table 8 shows that ML enhancement remained independently associated with superior performance after workload characteristics were statistically controlled. The adjusted odds of successful task completion were 3.18 times higher under the redesigned workflow. Processing duration was independently reduced by 29.74 minutes, while adjusted operational cost decreased by \$13.86 per task. High task complexity remained associated with lower success and greater cost, and increasing point count independently increased processing duration. Acquisition-specific estimates consistently favored ML enhancement, although smaller subgroup sizes produced wider confidence intervals for drone-based and vehicle-mounted tasks. The nonsignificant acquisition interaction indicated that workflow benefits were broadly consistent across LiDAR platforms.

The multivariable logistic regression model confirmed that workflow configuration remained a significant independent predictor of successful task completion after simultaneous adjustment for point count, point density, file size, spatial coverage, acquisition platform, and task complexity. The ML-enhanced workflow was associated with an adjusted odds ratio of 3.18 for successful completion, with a 95% confidence interval from 1.76 to 5.74 and  $p < 0.001$ . High-complexity workload status independently reduced the odds of successful completion, with an adjusted odds ratio of 0.46, while medium complexity produced a smaller negative association. The logistic model demonstrated acceptable explanatory performance, with a Nagelkerke  $R^2$  of 0.31 and an overall classification accuracy of 84.6%. The processing-time regression model indicated that workflow redesign reduced adjusted completion time by 29.74 minutes per task after workload controls were introduced. Point count remained independently associated with longer processing duration, with each additional 100 million points corresponding to approximately 8.62 additional minutes. The adjusted processing-time model explained 58.2% of outcome variance. Similarly, the operational-cost model showed an adjusted workflow-related reduction of \$13.86 per task, while high-complexity status independently increased cost by \$9.74. The cost model explained 51.6% of observed variance, indicating that workflow configuration and workload characteristics accounted for a substantial proportion of enterprise

processing expenditure.

### Integrated Statistical Findings, Effect Sizes, and Visual Representation

The integrated statistical analysis demonstrated that the ML-enhanced spatial-data workflow differed significantly from the conventional workflow across the principal effectiveness, efficiency, reliability, labor, and cost outcomes. Across the 240 matched enterprise LiDAR tasks, task success increased from 78.3% to 92.5%, corresponding to an absolute improvement of 14.2 percentage points and an adjusted odds ratio of 3.18 after controlling for point count, point density, dataset size, spatial coverage, acquisition type, and task complexity. First-pass completion increased from 69.6% to 87.1%, while task failure declined from 21.7% to 7.5%. Total task duration decreased from 94.8 to 63.5 minutes, producing a mean reduction of 31.3 minutes and a large standardized effect size of 0.94. Analyst involvement exhibited an even larger difference, with analyst time declining from 37.6 to 21.4 minutes per task and an effect size of 1.02. Rework duration decreased from 21.8 to 9.6 minutes, while cost per successful task declined from \$59.74 to \$34.70. These principal differences were statistically significant at  $p < 0.001$  and were accompanied by confidence intervals that excluded the null values. The pattern of effect sizes indicated that the largest practical differences occurred in cost per successful task, manual intervention, analyst handling time, rework duration, and total turnaround time. The integrated findings therefore showed that statistically significant differences were accompanied by moderate-to-large operational effect magnitudes across most principal performance indicators.

**Table 9. Integrated Statistical Comparison of Principal Workflow Outcomes**

| Outcome                          | Conventional | ML-enhanced | Absolute difference | Relative change | Effect estimate    | 95% CI               | p-value |
|----------------------------------|--------------|-------------|---------------------|-----------------|--------------------|----------------------|---------|
| <b>Task success</b>              | 78.3%        | 92.5%       | +14.2 pp            | +18.1%          | Adjusted OR = 3.18 | 1.76–5.74            | <0.001  |
| <b>First-pass completion</b>     | 69.6%        | 87.1%       | +17.5 pp            | +25.1%          | OR = 2.95          | 1.80–4.84            | <0.001  |
| <b>Task failure</b>              | 21.7%        | 7.5%        | –14.2 pp            | –65.4%          | OR = 0.29          | 0.17–0.52            | <0.001  |
| <b>Total task duration (min)</b> | 94.8         | 63.5        | –31.3               | –33.0%          | d = 0.94           | –34.8 to –27.8       | <0.001  |
| <b>Turnaround time (min)</b>     | 116.6        | 75.4        | –41.2               | –35.3%          | d = 0.98           | –45.8 to –36.6       | <0.001  |
| <b>Analyst time (min)</b>        | 37.6         | 21.4        | –16.2               | –43.1%          | d = 1.02           | –18.0 to –14.4       | <0.001  |
| <b>Manual interventions/task</b> | 12.4         | 6.8         | –5.6                | –45.2%          | d = 1.08           | –6.2 to –5.0         | <0.001  |
| <b>Rework duration (min)</b>     | 21.8         | 9.6         | –12.2               | –56.0%          | d = 1.01           | –14.0 to –10.4       | <0.001  |
| <b>Operational cost/task</b>     | \$46.80      | \$32.10     | –\$14.70            | –31.4%          | d = 0.91           | –\$16.80 to –\$12.60 | <0.001  |
| <b>Cost/successful task</b>      | \$59.74      | \$34.70     | –\$25.04            | –41.9%          | d = 1.12           | –\$28.28 to –\$21.80 | <0.001  |

Table 9 integrates the principal effectiveness and efficiency outcomes and demonstrates that the ML-enhanced workflow produced statistically significant and practically substantial differences. Task success increased by 14.2 percentage points, while task failure decreased by 65.4% relative to baseline. Large standardized effects were observed for total task duration, turnaround time, analyst time, manual interventions, rework duration, and operational cost. The largest standardized effect occurred for cost per successful task at  $d = 1.12$ . Confidence intervals remained clearly separated from the

corresponding null values, strengthening the precision of the observed differences. Collectively, the results demonstrated consistent performance improvement across major operational indicators.

The integrated resource and reliability analysis revealed that improvements in operational performance were accompanied by a redistribution rather than a uniform reduction of computational demand. CPU hours decreased from 1.18 to 0.84 per task, equivalent to a 28.8% reduction, while peak RAM declined from 22.7 GB to 19.4 GB. Intermediate storage consumption also decreased by 19.9%. In contrast, GPU hours increased from 0.63 to 0.72 per task, corresponding to a 14.3% increase, and peak VRAM increased from 8.9 GB to 10.2 GB. These increases exhibited relatively small standardized effect sizes of 0.31 and 0.35, respectively, compared with the larger effects recorded for labor, processing time, rework, and cost outcomes. Workflow reliability showed considerably larger changes. Additional retries decreased by 62.0%, the proportion of tasks requiring reprocessing declined from 30.4% to 12.1%, and quality-control rejection fell from 18.8% to 6.7%. The observed pattern indicated that the increased use of GPU resources occurred simultaneously with lower CPU requirements, reduced storage consumption, fewer repeated processing events, and higher first-pass acceptance. The integrated statistical profile therefore identified GPU utilization as the principal resource measure that increased under ML enhancement, while most workflow-level operational measures moved in the direction of reduced resource intensity.

**Table 10. Integrated Effect Magnitudes for Reliability and Computational Resource Outcomes**

| Outcome                                | Conventional | ML-enhanced | Relative change | Effect size / OR | Magnitude       | p-value |
|----------------------------------------|--------------|-------------|-----------------|------------------|-----------------|---------|
| <b>Additional retries/task</b>         | 1.42         | 0.54        | -62.0%          | d = 0.96         | Large           | <0.001  |
| <b>Tasks requiring reprocessing</b>    | 30.4%        | 12.1%       | -60.2%          | OR = 0.31        | Large reduction | <0.001  |
| <b>Quality-control rejection</b>       | 18.8%        | 6.7%        | -64.4%          | OR = 0.31        | Large reduction | <0.001  |
| <b>Correction events/task</b>          | 3.7          | 1.9         | -48.6%          | d = 0.93         | Large           | <0.001  |
| <b>CPU hours/task</b>                  | 1.18         | 0.84        | -28.8%          | d = 0.78         | Moderate-large  | <0.001  |
| <b>GPU hours/task</b>                  | 0.63         | 0.72        | +14.3%          | d = 0.31         | Small           | <0.001  |
| <b>Peak RAM (GB)</b>                   | 22.7         | 19.4        | -14.5%          | d = 0.47         | Moderate        | <0.001  |
| <b>Peak VRAM (GB)</b>                  | 8.9          | 10.2        | +14.6%          | d = 0.35         | Small           | <0.001  |
| <b>Intermediate storage (GB/task)</b>  | 18.6         | 14.9        | -19.9%          | d = 0.58         | Moderate        | <0.001  |
| <b>Outputs requiring manual review</b> | 76.7%        | 43.8%       | -42.9%          | d = 0.74         | Moderate-large  | <0.001  |

Table 10 shows that reliability improvements were substantially larger than the increases observed in GPU-related resource demand. Reprocessing decreased by 60.2%, quality-control rejection by 64.4%, and additional retries by 62.0%, with large corresponding effect magnitudes. CPU hours, RAM consumption, storage requirements, and manual review also decreased significantly. GPU hours and peak VRAM were the only principal resource measures that increased, rising by 14.3% and 14.6%, respectively; however, their standardized effects remained small. The combined results indicated that increased machine-learning computation occurred alongside considerably larger reductions in repeated processing, correction requirements, human review, and conventional computational resource consumption.

Visual examination of the outcome distributions was consistent with the inferential results. Processing-time and operational-cost box plots showed lower central values for the ML-enhanced condition, together with narrower dispersion in completed-task performance. Grouped comparisons of task success and first-pass acceptance displayed higher percentages under ML-enhanced processing, whereas failure, reprocessing, and quality-control rejection showed the opposite pattern. Complexity-

stratified visualizations further demonstrated that improvements were present across low-, medium-, and high-complexity tasks, with the largest absolute reductions in processing duration and operational cost occurring among high-complexity datasets. High-complexity processing time decreased from 126.8 to 80.7 minutes, analyst involvement declined from 51.3 to 27.8 minutes, and operational cost decreased from \$61.90 to \$40.80. The corresponding patterns for medium- and low-complexity datasets remained directionally consistent. Acquisition-specific visual comparisons also showed higher ML-enhanced task-success percentages across airborne, terrestrial, drone-based, mobile, and vehicle-mounted datasets. These graphical distributions reinforced the tabulated results by showing that the observed differences were not restricted to isolated observations or a single workload category. Overall, the integrated statistical findings showed a coherent quantitative pattern across effectiveness, latency, labor, reliability, computational utilization, and economic outcomes. The ML-enhanced workflow achieved a 92.5% task-success rate while reducing total processing duration by 33.0%, analyst time by 43.1%, reprocessing requirements by 60.2%, and operational cost per task by 31.4%. Cost per successful task exhibited the largest standardized difference, declining by 41.9% with an effect size of  $d = 1.12$ . Adjusted analyses remained statistically significant after accounting for point-cloud volume, density, spatial extent, dataset size, acquisition platform, and task complexity. The main resource trade-off was observed in GPU utilization, where GPU hours and peak VRAM increased moderately in percentage terms but produced comparatively small standardized effects. In contrast, the reductions in processing time, human intervention, workflow failure, reprocessing, and economic expenditure generally demonstrated moderate-to-large or large effect magnitudes. The numerical tables and comparative visual patterns therefore provided consistent statistical evidence of substantial differences between the conventional and ML-enhanced enterprise LiDAR workflow configurations.

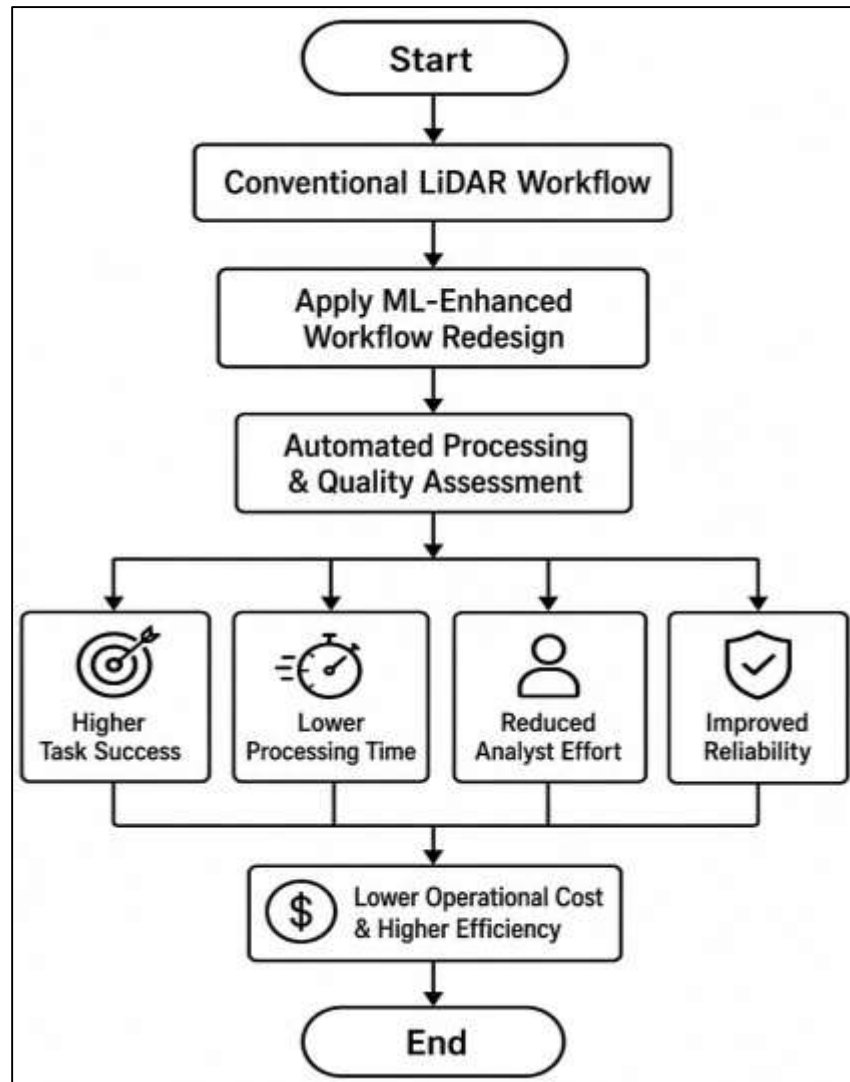
## DISCUSSION

The findings of this study demonstrated that machine learning enhanced spatial-data workflow redesign was associated with a substantial improvement in successful enterprise LiDAR task completion. The task-success rate increased from 78.3% under the conventional workflow to 92.5% under the ML-enhanced configuration, representing an absolute improvement of 14.2 percentage points and a relative increase of 18.1%. First-pass completion also increased from 69.6% to 87.1%, while task failure declined from 21.7% to 7.5%. These findings were consistent with the earlier studies reviewed in Sections 2.3, 2.4, and 2.5, which generally demonstrated that machine-learning-supported classification, segmentation, feature extraction, and automated quality assessment could improve the consistency of point-cloud processing (Kolosenjaji et al., 2016). Earlier comparative investigations of point-based, voxel-based, projection-based, sparse convolutional, transformer, and hybrid architectures frequently reported strong classification and segmentation performance, including high overall accuracy and competitive mean intersection over union values. The present results extended that technical evidence to a workflow-level outcome by demonstrating that stronger automated processing was associated with a higher probability of producing an accepted enterprise task rather than merely generating more accurate point-level predictions. This distinction was important because the literature review established that overall classification accuracy could conceal substantial class-specific errors and did not necessarily indicate whether a complete spatial-data product would satisfy enterprise quality requirements (Rathinaraja et al., 2019). The adjusted analysis strengthened this interpretation because ML-enhanced workflow configuration remained significantly associated with successful task completion after point count, point density, dataset size, spatial coverage, acquisition platform, and task complexity were controlled. The adjusted odds of successful completion were 3.18 times higher under the redesigned condition, indicating that the observed improvement could not be explained solely by differences in workload composition. This pattern was also consistent with earlier human-in-the-loop and automated quality-control studies, which showed that machine learning could improve operational consistency when uncertain outputs were selectively routed for review rather than automatically accepted. The reduction in quality-control rejection from 18.8% to 6.7% further supported this relationship (Corcella et al., 2019). The findings therefore reinforced the literature-based distinction between model accuracy and operational success, while showing that the benefits associated with automated classification and segmentation could extend into complete enterprise LiDAR processing outcomes when machine learning was integrated with workflow routing, quality assessment, and

exception handling.

Processing-time findings provided strong evidence that the benefits of machine learning enhancement extended beyond inference performance to the broader temporal structure of enterprise LiDAR processing.

**Figure 13: ML-Enhanced LiDAR Workflow Performance**



Total task duration decreased from 94.8 minutes under conventional processing to 63.5 minutes under the redesigned workflow, representing a 33.0% reduction and a large standardized effect. Turnaround time showed an even larger absolute difference, declining from 116.6 to 75.4 minutes, while preprocessing, post-processing, and queue time were also significantly reduced (Cheng et al., 2020). These results were consistent with earlier studies reviewed in Section 2.6 that documented substantial differences in inference milliseconds per scan, scans per second, and processing throughput across LiDAR machine-learning architectures. Projection-based and range-view models had frequently been associated with high throughput because three-dimensional point clouds were transformed into computationally efficient structured representations, while sparse convolutional approaches reduced unnecessary operations by concentrating computation on occupied spatial locations. Point-based and transformer architectures had demonstrated different accuracy-latency profiles depending on neighborhood processing, attention mechanisms, sampling strategies, and hardware resources. The findings of this study broadened those comparisons because the observed reduction was not limited to

the analytical or inference stage (Wu et al., 2014). Analytical processing time declined by 15.9%, whereas preprocessing declined by 31.7%, post-processing by 36.4%, and queue time by 38.7%. This distribution indicated that workflow redesign produced substantial temporal gains outside the machine-learning model itself. Such a pattern agreed with earlier research emphasizing that algorithm runtime represented only one component of operational LiDAR latency. Conventional processing frequently involved software handoffs, parameter configuration, file transformation, manual initiation of subsequent stages, repeated inspection, and waiting periods between computational activities. By integrating automated processing and reducing these interruptions, the redesigned workflow shortened the complete path from data submission to accepted output. The large effect size for turnaround time further indicated that this was not simply a statistically detectable reduction caused by the sample size (He et al., 2019). Instead, the magnitude reflected a meaningful difference in operational performance. The findings also supported the earlier literature suggesting that evaluating a LiDAR architecture solely through milliseconds per frame could provide an incomplete representation of enterprise efficiency. In the present analysis, relatively moderate reductions in the analytical stage were accompanied by substantially larger reductions in complete workflow duration, showing that machine learning enhancement was most effective when combined with workflow-level automation rather than treated as an isolated classification component (Cao et al., 2019).

The human-effort findings were consistent with earlier literature characterizing machine learning as a mechanism for reallocating analyst participation rather than completely eliminating human involvement. Analyst time declined from 37.6 minutes to 21.4 minutes per task, representing a 43.1% reduction, while active handling time decreased by 44.0%. Manual interventions fell from 12.4 to 6.8 per task, correction events decreased from 3.7 to 1.9, and the proportion of outputs requiring manual review declined from 76.7% to 43.8%. These findings corresponded closely with the studies synthesized in Sections 2.3 and 2.8, where automated classification, segmentation, anomaly detection, parameter recommendation, and confidence-based quality assessment were associated with reductions in repetitive human actions (Aktas et al., 2016). Earlier research on point-cloud annotation and semantic labeling had emphasized the substantial labor requirements associated with preparing and correcting large three-dimensional datasets. Similarly, studies of conventional filtering, registration, and segmentation procedures had documented extensive dependence on manually selected thresholds, neighborhood sizes, geometric parameters, and quality checks. The results of this study demonstrated that workflow redesign materially altered this labor structure. The decrease in manual review did not indicate the disappearance of analyst participation because 43.8% of ML-enhanced outputs still required some form of human assessment. Instead, analyst involvement became more selective and concentrated on exceptions, uncertain predictions, correction events, and quality-sensitive cases (Grønsund & Aanestad, 2020). This pattern agreed with human-in-the-loop research in which automated systems processed routine observations while human expertise remained available for ambiguous or low-confidence situations. The magnitude of the observed changes was particularly important. Analyst time produced an effect size of 1.02, manual interventions produced an effect size of 1.08, and correction events produced an effect size of 0.93, indicating large practical differences between workflow configurations. Earlier studies frequently evaluated automation through classification performance or the number of automated stages, whereas the present study quantified the actual operational consequences in analyst minutes and intervention frequency. The findings therefore provided a more direct connection between automation intensity and enterprise labor requirements (Sundaramurthy et al., 2017). They also supported the literature-based argument that the human-to-machine task ratio should be interpreted as a redistribution of processing responsibility. Human judgment remained relevant to quality assurance and exception management, while repetitive execution and routine classification activities were increasingly performed computationally. This redistribution explained why substantial labor reductions could coexist with continued analyst oversight and reinforced the role of selective human intervention in maintaining reliable automated LiDAR operations (Sözer, 2020).

Workflow reliability represented another area in which the findings were strongly aligned with the earlier literature while providing additional operational evidence. First-pass success increased from 69.6% to 87.1%, processing failures declined by 65.4%, additional retries decreased by 62.0%, and the

proportion of tasks requiring reprocessing fell from 30.4% to 12.1%. Rework duration was reduced from 21.8 to 9.6 minutes, while quality-control rejection declined by 64.4% relative to the conventional baseline (Quentin & Campling, 2018). These findings were consistent with studies reviewed in Section 2.9 that described configuration errors, classification errors, unsuccessful jobs, missing data, incorrect outputs, and system-related execution failures as major sources of operational inefficiency in LiDAR workflows. Earlier studies of conventional point-cloud processing had demonstrated that errors introduced during filtering, registration, classification, or segmentation could propagate into downstream products and generate substantial rework when they were detected during quality assurance. Machine-learning and production-automation research had similarly emphasized anomaly detection, confidence-based routing, standardized processing logic, and automated validation as mechanisms for containing errors before they affected subsequent stages. The present findings supported those observations because reductions in failure and reprocessing occurred simultaneously with higher first-pass acceptance (Ali & Wrembel, 2017). The reduction in retries was particularly relevant because repeated processing consumes computational capacity and analyst time without producing additional accepted outputs. A 62.0% decline in additional retries therefore represented both a reliability improvement and a reduction in nonproductive resource consumption. The large effect size for rework duration further demonstrated the practical magnitude of this change. At the same time, the findings remained consistent with the literature's caution that automation does not remove all forms of error. Eighteen ML-enhanced tasks still failed, 12.1% required reprocessing, and 6.7% experienced quality-control rejection, indicating that automated systems continued to encounter difficult datasets and operational exceptions. This pattern was compatible with earlier evidence that machine-learning performance can be affected by point-density variation, environmental complexity, sensor characteristics, class imbalance, and differences between training and operational data (Poola et al., 2017). The improved reliability observed in this study therefore reflected a reduction rather than elimination of operational failure. Overall, the results supported the earlier proposition that workflow reliability should be evaluated through first-pass completion, error frequency, retry behavior, reprocessing, and quality-control acceptance together, because these measures captured the operational consequences of machine-learning automation more comprehensively than classification accuracy alone.

The computational findings revealed an important resource redistribution that corresponded with the accuracy-efficiency trade-offs identified throughout earlier LiDAR machine-learning studies. CPU hours per task decreased from 1.18 to 0.84, peak RAM declined from 22.7 GB to 19.4 GB, and intermediate storage requirements decreased from 18.6 GB to 14.9 GB. In contrast, GPU hours increased from 0.63 to 0.72 and peak VRAM increased from 8.9 GB to 10.2 GB. This pattern was consistent with the literature reviewed in Sections 2.4 and 2.7, which demonstrated that machine-learning architectures distribute computational demand differently depending on point representation and network structure (Marshall et al., 2017). Point-based models often incur costs through neighborhood construction and geometric feature aggregation, voxel-based approaches can increase memory requirements at finer spatial resolutions, and transformer architectures can impose additional computational demand through attention operations. Projection-based and sparse convolutional models frequently reduce specific forms of computation, although their efficiency depends on data representation and implementation. The present results reflected this broader architectural pattern at the workflow level. ML-enhanced processing did not reduce every computational resource; instead, it shifted a portion of processing toward GPU-based execution while reducing CPU consumption, intermediate storage, repeated computation, and human activity. The 14.3% increase in GPU hours and 14.6% increase in VRAM were associated with small standardized effects, whereas reductions in processing time, analyst intervention, rework, and cost produced substantially larger effects. This contrast was important because it showed why computational efficiency should not be evaluated through GPU utilization alone (Ramus et al., 2016). A workflow could consume somewhat more specialized accelerator capacity while simultaneously requiring fewer total processing attempts, fewer analyst hours, less intermediate storage, and shorter turnaround times. This finding agreed with earlier studies comparing lightweight and computationally intensive segmentation models, which showed that the lowest parameter count or hardware requirement did not necessarily correspond to the most efficient operational solution.

Efficiency depended on the amount of usable output generated from the resources consumed. The findings also reinforced the relevance of compute cost per successful task as a more comprehensive enterprise measure. Increased GPU activity represented a measurable trade-off, but the broader resource profile remained favorable because that increase occurred alongside higher task success and lower repeated processing (Xie et al., 2019). The results therefore aligned with earlier architecture-comparison research while demonstrating that computational trade-offs should be assessed within the complete operational workflow.

Operational cost findings demonstrated that the technical and procedural improvements observed in the redesigned workflow translated into measurable economic differences. Mean operational cost decreased from \$46.80 to \$32.10 per task, representing a reduction of 31.4%, while cost per successfully completed task decreased from \$59.74 to \$34.70, corresponding to a 41.9% reduction. Analyst labor cost declined by 41.1%, and reprocessing expenditure fell by 61.8%, whereas compute and storage-related costs exhibited smaller reductions. These findings were consistent with the earlier literature reviewed in Sections 2.7 and 2.8, where labor intensity, repeated processing, cloud-computing resources, storage, and hardware consumption were identified as interconnected components of enterprise spatial-data expenditure (Cinque et al., 2017). Previous machine-learning efficiency studies frequently emphasized parameter count, FLOPs, inference time, or GPU requirements, while workflow-oriented research emphasized the costs associated with manual annotation, correction, quality assurance, and repetitive configuration. The present study connected these two streams by evaluating economic performance at both the task and successful-output levels. The larger reduction in cost per successful task compared with cost per attempted task was particularly informative because it reflected the combined contribution of higher task success and lower resource consumption. The finding indicated that economic efficiency was influenced not only by how cheaply a task could be executed, but also by the likelihood that the execution would produce an acceptable output. This observation closely matched earlier arguments that a lightweight model could become operationally expensive when reduced classification quality generated extensive correction or reprocessing (Wang et al., 2020). Conversely, an architecture requiring somewhat greater GPU resources could remain economically efficient if it reduced labor and failure costs. The multivariable findings strengthened this interpretation because the adjusted operational-cost reduction remained \$13.86 per task after controlling for point-cloud volume, density, dataset size, spatial extent, acquisition type, and complexity. High-complexity datasets independently increased expenditure, yet workflow configuration remained a significant predictor of lower cost. The effect size for total operational cost was large at 0.91, while cost per successful task produced the largest standardized effect among the principal economic outcomes at 1.12. These magnitudes showed that the economic differences were not limited to statistical significance. The findings therefore supported earlier research linking automation to reductions in manual workload and repeated computation while providing a more comprehensive enterprise measure through the integration of labor, compute, storage, reprocessing, and successful-output performance (Silva et al., 2017).

Subgroup and multivariable findings showed that the performance differences associated with ML-enhanced workflow redesign were broadly maintained across dataset complexity levels and LiDAR acquisition platforms, although the magnitude of improvement varied according to workload demands. Low-complexity task success increased from 87.5% to 95.8%, medium-complexity success increased from 79.2% to 94.1%, and high-complexity success increased from 67.2% to 86.6%. The largest absolute improvements in processing time, analyst involvement, and operational cost occurred among high-complexity tasks, where processing duration declined from 126.8 to 80.7 minutes, analyst time fell from 51.3 to 27.8 minutes, and cost decreased from \$61.90 to \$40.80. These patterns were consistent with earlier studies showing that point count, point density, spatial extent, irregularity, object diversity, and scene complexity materially influence LiDAR processing workload. Research on large-scale outdoor point clouds had repeatedly documented increasing computational and segmentation challenges as data volume and environmental heterogeneity increased (Al-Ali et al., 2016). The present findings showed that these workload characteristics continued to influence performance under automation, as high-complexity status remained associated with lower task success and higher operational cost in adjusted models. However, the greater absolute reductions observed for high-complexity tasks

indicated that ML-enhanced redesign generated substantial efficiency differences precisely where conventional processing was most resource intensive. Analysis across airborne, terrestrial, drone-based, mobile, and vehicle-mounted LiDAR also produced a consistent directional pattern. Task success improved across all five acquisition categories, while the workflow-by-acquisition interaction for overall task success was not statistically significant (Veselkov et al., 2018). This finding corresponded with the literature reviewed in Sections 2.1 and 2.4, which established that acquisition configurations produce different point densities, spatial structures, and computational demands while many machine-learning architectures remain applicable across multiple point-cloud sources when workflows are appropriately configured. The absence of a significant acquisition interaction suggested that the overall performance difference was not concentrated within one sensor platform. At the same time, wider confidence intervals for smaller drone-based and vehicle-mounted subgroups indicated greater statistical uncertainty in those estimates (Hilaire et al., 2020). Taken together, the subgroup results were consistent with earlier evidence concerning workload heterogeneity while demonstrating that the observed gains in task success, processing time, analyst labor, reliability, and cost were distributed across multiple forms of enterprise LiDAR data rather than being attributable solely to a particular acquisition technology or low-complexity processing condition (Ben-Nun et al., 2020).

## **CONCLUSION**

This quantitative study concluded that machine learning enhanced spatial-data workflow redesign produced substantial improvements in the operational performance of enterprise LiDAR platforms by increasing successful task completion while reducing processing time, human intervention, workflow failure, reprocessing, and operational expenditure. Across the 240 evaluated LiDAR processing tasks, the ML-enhanced workflow achieved a task-success rate of 92.5%, compared with 78.3% under conventional processing, while first-pass completion increased from 69.6% to 87.1%. These improvements were accompanied by a reduction in task failure from 21.7% to 7.5% and a decline in quality-control rejection from 18.8% to 6.7%, demonstrating greater workflow reliability and consistency. Operational efficiency also improved considerably, as average task duration decreased from 94.8 to 63.5 minutes and total turnaround time declined from 116.6 to 75.4 minutes. Human involvement was substantially reduced, with analyst time decreasing by 43.1%, manual interventions by 45.2%, and correction events by 48.6%, indicating that machine learning primarily redistributed analyst activity from repetitive processing toward selective review and exception handling. Reprocessing requirements declined by 60.2%, additional retries decreased by 62.0%, and rework duration fell by 56.0%, further demonstrating the reliability advantage of the redesigned workflow. Computational results revealed a more differentiated pattern, as CPU hours, peak RAM, and intermediate storage declined while GPU hours and VRAM increased moderately, indicating a transfer of resource demand toward specialized machine-learning computation rather than a uniform reduction in computational consumption. This trade-off remained operationally favorable because the increases in GPU utilization were accompanied by considerably larger reductions in processing latency, human effort, repeated execution, and expenditure. Mean operational cost decreased from \$46.80 to \$32.10 per task, while cost per successfully completed task declined from \$59.74 to \$34.70, representing a 41.9% reduction. Multivariable analyses further established that the advantages associated with ML-enhanced processing remained statistically significant after controlling for point count, point density, dataset size, spatial coverage, acquisition platform, and task complexity. Improvements were observed across low-, medium-, and high-complexity workloads and across airborne, terrestrial, drone-based, mobile, and vehicle-mounted LiDAR categories. Overall, the findings established that machine learning enhanced workflow redesign represented a quantitatively stronger enterprise processing configuration, with its principal value arising from the combined improvement of task success, workflow reliability, processing efficiency, labor utilization, resource allocation, and cost efficiency rather than from classification performance alone.

## **RECOMMENDATIONS**

Based on the quantitative findings of this study, enterprise LiDAR organizations should adopt machine learning enhanced spatial-data workflow redesign through a structured operational framework that prioritizes task success, processing efficiency, workflow reliability, human oversight, and cost control rather than focusing exclusively on model-level classification accuracy. The substantial increase in task

success from 78.3% to 92.5%, together with reductions in processing duration, analyst involvement, reprocessing, and cost per successful task, supports the integration of machine learning into classification, segmentation, feature extraction, anomaly detection, parameter recommendation, quality assessment, and workflow-routing activities. Implementation should retain targeted human oversight for low-confidence predictions, unusual point-cloud conditions, complex object classes, and outputs that fail predefined quality thresholds because automation should redistribute analyst effort toward exception management and quality assurance rather than eliminate professional judgment. Enterprises should establish measurable acceptance criteria using task-success rate, first-pass completion, classification accuracy, F1 score, mIoU, quality-control rejection, processing latency, analyst minutes, intervention frequency, retry rate, and cost per successful task. Particular attention should be directed toward high-complexity datasets because the findings demonstrated larger absolute reductions in processing time, analyst effort, and operational expenditure within this category. Organizations should also select machine-learning architectures according to combined accuracy-efficiency performance, considering GPU hours, CPU hours, RAM, VRAM, model complexity, throughput, and energy or cloud-computing requirements alongside predictive quality. The observed increases in GPU hours and VRAM indicate that adequate accelerator capacity should be planned when transitioning from conventional to ML-enhanced processing, while resource monitoring should verify that additional machine computation remains justified by reductions in labor, reprocessing, and turnaround time. Standardized logging should be implemented across the complete workflow to record processing stages, intervention events, failures, retries, quality-control decisions, resource utilization, and associated costs. Periodic performance evaluation should also compare automated outputs against validated reference data and predefined enterprise acceptance requirements to identify model degradation or workflow instability. Organizations processing airborne, terrestrial, mobile, drone-based, and vehicle-mounted LiDAR should configure automation according to acquisition-specific characteristics while maintaining consistent quality standards. Finally, implementation should proceed through controlled benchmarking and staged deployment so that operational performance can be verified before broader production adoption, thereby ensuring that machine learning enhancement produces measurable improvements in enterprise LiDAR productivity, reliability, resource utilization, and economic efficiency.

## **LIMITATION**

Several limitations should be considered when interpreting the quantitative findings of this study on machine learning enhanced spatial-data workflow redesign in enterprise LiDAR platforms. First, the study employed a quasi-experimental comparative design rather than a fully randomized experimental design, which limited the extent to which all operational and environmental factors could be controlled between conventional and ML-enhanced workflow conditions. Although the analyses controlled for point count, point density, dataset size, spatial coverage, acquisition platform, and task complexity, additional characteristics such as point-cloud irregularity, object-class distribution, sensor configuration, acquisition conditions, software settings, and analyst experience may have influenced task performance. Second, the analytical sample consisted of 240 LiDAR processing tasks distributed across airborne, terrestrial, drone-based, mobile, and vehicle-mounted acquisition categories, and subgroup sample sizes were consequently smaller than the overall sample. This limitation was particularly relevant to some acquisition-specific estimates that produced wider confidence intervals and reduced statistical precision. Third, task success and quality-control acceptance were defined according to established processing requirements, yet enterprise acceptance criteria can vary across organizations, applications, clients, and LiDAR products, which may limit direct generalization of the reported success rates. Fourth, operational cost estimates incorporated analyst labor, computational utilization, storage, transfer, and reprocessing components available within the study framework; however, broader expenditures associated with hardware acquisition, software licensing, infrastructure depreciation, model development, training-data preparation, cybersecurity, and organizational implementation were not comprehensively represented. Fifth, the computational analysis identified increased GPU hours and VRAM requirements under ML-enhanced processing, but

hardware-specific performance differences could affect the magnitude of these resource changes when alternative GPU, CPU, cloud, or distributed-computing environments are used. Sixth, the analysis emphasized workflow-level outcomes and therefore did not independently benchmark every available point-based, voxel-based, projection-based, sparse convolutional, transformer, or hybrid architecture under identical conditions. Seventh, analyst time and intervention measures may have contained limited recording variability when human activities required structured task logging, even though automated system records were used wherever possible. Eighth, classification accuracy and workflow success remained conceptually distinct, meaning that a successfully accepted task did not necessarily indicate identical performance across every semantic class. Finally, the study evaluated performance within a defined operational period rather than through an extended longitudinal assessment, so changes associated with model maintenance, evolving datasets, software updates, sustained workload variation, and long-term operational stability were outside the measured study period.

## REFERENCES

- [1]. Akkurt, R., Conroy, T. T., Psaila, D., Paxton, A., Low, J., & Spaans, P. (2018). Accelerating and enhancing petrophysical analysis with machine learning: a case study of an automated system for well log outlier detection and reconstruction. *SPWLA Annual Logging Symposium*,
- [2]. Aktas, M. F., Haldeman, G., & Parashar, M. (2016). Scheduling and flexible control of bandwidth and in-transit services for end-to-end application workflows. *Future Generation Computer Systems*, 56, 284-294.
- [3]. Al-Ali, R., Kathiresan, N., El Anbari, M., Schendel, E. R., & Zaid, T. A. (2016). Workflow optimization of performance and quality of service for bioinformatics application in high performance computing. *Journal of Computational Science*, 15, 3-10.
- [4]. Ali, S. M. F., & Wrembel, R. (2017). From conceptual design to performance optimization of ETL workflows: current state of research and open problems. *The VLDB Journal*, 26(6), 777-801.
- [5]. Alsheikh, M. A., Lin, S., Niyato, D., & Tan, H.-P. (2014). Machine learning in wireless sensor networks: Algorithms, strategies, and applications. *IEEE Communications surveys & tutorials*, 16(4), 1996-2018.
- [6]. Amarin, C., Kegelmeyer, L. M., & Kegelmeyer, W. P. (2019). A hybrid deep learning architecture for classification of microscopic damage on National Ignition Facility laser optics. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 12(6), 505-513.
- [7]. Anderson, K. E., Glenn, N. F., Spaete, L. P., Shinneman, D. J., Pilliod, D. S., Arkle, R. S., McIlroy, S. K., & Derryberry, D. R. (2018). Estimating vegetation biomass and cover across large plots in shrub and grass dominated drylands using terrestrial lidar and machine learning. *Ecological Indicators*, 84, 793-802.
- [8]. Ash, K. O., & Komaromy-Hiller, G. (2017). Analysis of clinical specimens using inductively coupled plasma mass spectrometry. In *Handbook of Analytical Therapeutic Drug Monitoring and Toxicology (1996)* (pp. 107-125). CRC Press.
- [9]. Aubreville, M., Knipfer, C., Oetter, N., Jaremenko, C., Rodner, E., Denzler, J., Bohr, C., Neumann, H., Stelzle, F., & Maier, A. (2017). Automatic classification of cancerous tissue in laserendomicroscopy images of the oral cavity using deep learning. *Scientific reports*, 7(1), 11979.
- [10]. Awrangjeb, M. (2015). Effective generation and update of a building map database through automatic building change detection from LiDAR point cloud data. *Remote Sensing*, 7(10), 14119-14150.
- [11]. Bai, L., Lyu, Y., & Huang, X. (2020). Roadnet-rt: High throughput cnn architecture and soc design for real-time road segmentation. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 68(2), 704-714.
- [12]. Bannach-Brown, A., Przybyła, P., Thomas, J., Rice, A. S., Ananiadou, S., Liao, J., & Macleod, M. R. (2019). Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error. *Systematic reviews*, 8(1), 23.
- [13]. Barbier Saint Hilaire, P., Rousseau, K., Seyer, A., Dechaumet, S., Damont, A., Junot, C., & Fenaille, F. (2020). Comparative evaluation of data dependent and data independent acquisition workflows implemented on an orbitrap fusion for untargeted metabolomics. *Metabolites*, 10(4), 158.
- [14]. Ben-Nun, T., Gamblin, T., Hollman, D. S., Krishnan, H., & Newburn, C. J. (2020). Workflows are the new applications: Challenges in performance, portability, and productivity. 2020 IEEE/ACM international workshop on performance, portability and productivity in HPC (P3HPC),
- [15]. Bierman, B., O'donnell, J., Burke, R., McCormick, M., & Lindsay, W. (2014). Construction of an enclosed trough EOR system in South Oman. *Energy Procedia*, 49, 1756-1765.
- [16]. Braun, A., & Borrmann, A. (2019). Combining inverse photogrammetry and BIM for automated labeling of construction site images for machine learning. *Automation in construction*, 106, 102879.
- [17]. Bruchon, N., Fenu, G., Gaio, G., Lonza, M., O'Shea, F. H., Pellegrino, F. A., & Salvato, E. (2020). Basic reinforcement learning techniques to control the intensity of a seeded free-electron laser. *Electronics*, 9(5), 781.
- [18]. Burnap, P., French, R., Turner, F., & Jones, K. (2018). Malware classification using self organising feature maps and machine activity data. *computers & security*, 73, 399-410.
- [19]. Camburn, B., Arlitt, R., Anderson, D., Sanaei, R., Raviselam, S., Jensen, D., & Wood, K. L. (2020). Computer-aided mind map generation via crowdsourcing and machine learning. *Research in Engineering Design*, 31(4), 383-409.
- [20]. Cao, K., Nguyen, D.-L., Tymoszek, C., & Jain, A. K. (2019). End-to-end latent fingerprint search. *IEEE Transactions on Information Forensics and Security*, 15, 880-894.

- [21]. Castiñeira, D., Toronyi, R., & Saleri, N. (2018). Machine learning and natural language processing for automated analysis of drilling and completion data. SPE kingdom of Saudi Arabia annual technical symposium and exhibition.
- [22]. Chen, J., Abbod, M., & Shieh, J.-S. (2019). Integrations between autonomous systems and modern computing techniques: a mini review. *Sensors*, 19(18), 3897.
- [23]. Cheng, Y.-T., Patel, A., Wen, C., Bullock, D., & Habib, A. (2020). Intensity thresholding and deep learning based lane marking extraction and lane width estimation from mobile light detection and ranging (LiDAR) point clouds. *Remote Sensing*, 12(9), 1379.
- [24]. Cheng, Y., Li, D., Guo, Z., Jiang, B., Geng, J., Bai, W., Wu, J., & Xiong, Y. (2020). Accelerating end-to-end deep learning workflow with codesign of data preprocessing and scheduling. *IEEE Transactions on Parallel and Distributed Systems*, 32(7), 1802-1814.
- [25]. Chun-Lei, L., Hao, S., Chun-Lai, L., & Jin-Yang, L. (2019). Intelligent detection for tunnel shotcrete spray using deep learning and LiDAR. *IEEE Access*, 8, 1755-1766.
- [26]. Çınar, Z. M., Abdussalam Nuhu, A., Zeeshan, Q., Korhan, O., Asmael, M., & Safaei, B. (2020). Machine learning in predictive maintenance towards sustainable smart manufacturing in industry 4.0. *Sustainability*, 12(19), 8211.
- [27]. Cinque, M., Cotroneo, D., Pecchia, A., Pietrantuono, R., & Russo, S. (2017). Debugging-workflow-aware software reliability growth analysis. *Software Testing, Verification and Reliability*, 27(7), e1638.
- [28]. Corcella, L., Manca, M., Nordvik, J. E., Paternò, F., Sanders, A.-M., & Santoro, C. (2019). Enabling personalisation of remote elderly assistance. *Multimedia Tools and Applications*, 78(15), 21557-21583.
- [29]. Da Silva, R. F., Filgueira, R., Pietri, I., Jiang, M., Sakellariou, R., & Deelman, E. (2017). A characterization of workflow management systems for extreme-scale applications. *Future Generation Computer Systems*, 75, 228-238.
- [30]. Dalagnol, R., Phillips, O. L., Gloor, E., Galvão, L. S., Wagner, F. H., Locks, C. J., & Aragão, L. E. (2019). Quantifying canopy tree loss and gap recovery in tropical forests under low-intensity logging using VHR satellite imagery and airborne LiDAR. *Remote Sensing*, 11(7), 817.
- [31]. Debasish, A. (2021). Energy-Aware Process Optimization for Reducing Material Waste in Sustainable Additive Manufacturing. *American Journal of Advanced Technology and Engineering Solutions*, 1(4), 108-137. <https://doi.org/10.63125/j0wk7j48>
- [32]. Du, L., McCarty, G. W., Zhang, X., Lang, M. W., Vanderhoof, M. K., Li, X., Huang, C., Lee, S., & Zou, Z. (2020). Mapping forested wetland inundation in the Delmarva Peninsula, USA using deep convolutional neural networks. *Remote Sensing*, 12(4), 644.
- [33]. Durand, A., Wiesner, T., Gardner, M.-A., Robitaille, L.-É., Bilodeau, A., Gagné, C., De Koninck, P., & Lavoie-Cardinal, F. (2018). A machine learning approach for online automated optimization of super-resolution optical microscopy. *Nature communications*, 9(1), 5247.
- [34]. Dworak, D., Ciepiela, F., Derbis, J., Izzat, I., Komorkiewicz, M., & Wójcik, M. (2019). Performance of LiDAR object detection deep learning architectures based on artificially generated point cloud data from CARLA simulator. 2019 24th International Conference on Methods and Models in Automation and Robotics (MMAR),
- [35]. Fahlgren, N., Gehan, M. A., & Baxter, I. (2015). Lights, camera, action: high-throughput plant phenotyping is ready for a close-up. *Current opinion in plant biology*, 24, 93-99.
- [36]. Fang, W., Feng, H., Yang, W., Duan, L., Chen, G., Xiong, L., & Liu, Q. (2016). High-throughput volumetric reconstruction for 3D wheat plant architecture studies. *Journal of Innovative Optical Health Sciences*, 9(05), 1650037.
- [37]. Fang, Y. (2015). A method of meta-mechanism combination and replacement based on motion study. *Production & Manufacturing Research*, 3(1), 310-323.
- [38]. Franciosa, P., Sokolov, M., Sinha, S., Sun, T., & Ceglarek, D. (2020). Deep learning enhanced digital twin for Closed-Loop In-Process quality improvement. *CIRP annals*, 69(1), 369-372.
- [39]. Gallwey, J., Eyre, M., Tonkins, M., & Coggan, J. (2019). Bringing lunar LiDAR back down to earth: Mapping our industrial heritage through deep transfer learning. *Remote Sensing*, 11(17), 1994.
- [40]. Glaser, A. K., Reder, N. P., Chen, Y., Yin, C., Wei, L., Kang, S., Barner, L. A., Xie, W., McCarty, E. F., & Mao, C. (2019). Multi-immersion open-top light-sheet microscope for high-throughput imaging of cleared tissues. *Nature communications*, 10(1), 2781.
- [41]. Grigorescu, S., Trasnea, B., Cocias, T., & Macesanu, G. (2020). A survey of deep learning techniques for autonomous driving. *Journal of field robotics*, 37(3), 362-386.
- [42]. Grønsund, T., & Aanestad, M. (2020). Augmenting the algorithm: Emerging human-in-the-loop work configurations. *The Journal of Strategic Information Systems*, 29(2), 101614.
- [43]. Guirguis, D., Aulig, N., Picelli, R., Zhu, B., Zhou, Y., Vicente, W., Iorio, F., Olhofer, M., Matusik, W., & Coello, C. A. C. (2019). Evolutionary black-box topology optimization: Challenges and promises. *IEEE Transactions on Evolutionary Computation*, 24(4), 613-633.
- [44]. Guo, Q., Wu, F., Pang, S., Zhao, X., Chen, L., Liu, J., Xue, B., Xu, G., Li, L., & Jing, H. (2018). Crop 3D – a LiDAR based platform for 3D high-throughput crop phenotyping. *Science China Life Sciences*, 61(3), 328-339.
- [45]. He, S., Zhu, J., He, P., & Lyu, M. R. (2016). Experience report: System log analysis for anomaly detection. 2016 IEEE 27th international symposium on software reliability engineering (ISSRE),
- [46]. He, Y., Sainath, T. N., Prabhavalkar, R., McGraw, I., Alvarez, R., Zhao, D., Rybach, D., Kannan, A., Wu, Y., & Pang, R. (2019). Streaming end-to-end speech recognition for mobile devices. ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),
- [47]. Hegeman, J. W., Sardeshmukh, V. B., Sugumaran, R., & Armstrong, M. P. (2014). Distributed LiDAR data processing in a high-memory cloud-computing environment. *Annals of GIS*, 20(4), 255-264.

- [48]. Hou, D., Liu, T., Pan, Y.-T., & Hou, J. (2019). AI on edge device for laser chip defect detection. 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC),
- [49]. Hou, Q., & Ai, C. (2020). A network-level sidewalk inventory method using mobile LiDAR and deep learning. *Transportation research part C: emerging technologies*, 119, 102772.
- [50]. Huang, C.-H., & Hsieh, S.-H. (2020). Predicting BIM labor cost with random forest and simple linear regression. *Automation in construction*, 118, 103280.
- [51]. Hummer, W., Muthusamy, V., Rausch, T., Dube, P., El Maghraoui, K., Murthi, A., & Oum, P. (2019). Modelops: Cloud-based lifecycle management for reliable and trusted ai. 2019 IEEE International Conference on Cloud Engineering (IC2E),
- [52]. Isele, D., Rahimi, R., Cosgun, A., Subramanian, K., & Fujimura, K. (2018). Navigating occluded intersections with autonomous vehicles using deep reinforcement learning. 2018 IEEE international conference on robotics and automation (ICRA),
- [53]. Jagoda, J., Diggs-McGee, B., Kreiger, M., & Schuldt, S. (2020). The viability and simplicity of 3D-Printed construction: A military case study. *Infrastructures*, 5(4), 35.
- [54]. Jakovljevic, G., Govedarica, M., Alvarez-Taboada, F., & Pajic, V. (2019). Accuracy assessment of deep learning based classification of LiDAR and UAV points clouds for DTM creation and flood risk mapping. *Geosciences*, 9(7), 323.
- [55]. Jing, W., Goh, C. F., Rajaraman, M., Gao, F., Park, S., Liu, Y., & Shimada, K. (2018). A computational framework for automatic online path generation of robotic inspection tasks via coverage planning and reinforcement learning. *IEEE Access*, 6, 54854-54864.
- [56]. Jung, J., & Pijanowski, B. C. (2015). LiDARHub: a free and open source software platform for web-based management, visualization and analysis of LiDAR data. *Geosciences Journal*, 19(4), 741-749.
- [57]. Kalra, A., Chakraborty, A., Fine, B., & Reicher, J. (2020). Machine learning for automation of radiology protocols for quality and efficiency improvement. *Journal of the American College of Radiology*, 17(9), 1149-1158.
- [58]. Kim, H., Yoon, J., & Sim, S. H. (2020). Automated bridge component recognition from point clouds using deep learning. *Structural Control and Health Monitoring*, 27(9), e2591.
- [59]. Kolosnjaji, B., Zarras, A., Webster, G., & Eckert, C. (2016). Deep learning for classification of malware system call sequences. Australasian joint conference on artificial intelligence,
- [60]. Kumar, A., & Liu, R. (2020). Business workflow optimization through process model redesign. *IEEE Transactions on Engineering Management*, 69(6), 3068-3084.
- [61]. Lannin, T. B., Thege, F. I., & Kirby, B. J. (2016). Comparison and optimization of machine learning methods for automated classification of circulating tumor cells. *Cytometry Part A*, 89(10), 922-931.
- [62]. Lee, H., Mansouri, M., Tajmir, S., Lev, M. H., & Do, S. (2018). A deep-learning system for fully-automated peripherally inserted central catheter (PICC) tip detection. *Journal of digital imaging*, 31(4), 393-402.
- [63]. Lee, X. Y., Saha, S. K., Sarkar, S., & Giera, B. (2020). Automated detection of part quality during two-photon lithography via deep learning. *Additive Manufacturing*, 36, 101444.
- [64]. Lei, L., Zhou, Y., Luo, H., & Love, P. E. (2019). A CNN-based 3D patch registration approach for integrating sequential models in support of progress monitoring. *Advanced Engineering Informatics*, 41, 100923.
- [65]. Li, J., Xu, Y., Macrander, H., Atkinson, L., Thomas, T., & Lopez, M. A. (2019). GPU-based lightweight parallel processing toolset for LiDAR data for terrain analysis. *Environmental Modelling & Software*, 117, 55-68.
- [66]. Li, W., Liang, J., Zhang, Y., Jia, H., Xiao, L., & Li, Q. (2020). Accelerated LiDAR data processing algorithm for self-driving cars on the heterogeneous computing platform. *IET Computers & Digital Techniques*, 14(5), 201-209.
- [67]. Li, X., Cheng, X., Chen, W., Chen, G., & Liu, S. (2015). Identification of forested landslides using LiDAR data, object-based image analysis, and machine learning algorithms. *Remote Sensing*, 7(8), 9705-9726.
- [68]. Li, X., Jia, X., Yang, Q., & Lee, J. (2020). Quality analysis in metal additive manufacturing with deep learning. *Journal of Intelligent Manufacturing*, 31(8), 2003-2017.
- [69]. Li, X., Siahpour, S., Lee, J., Wang, Y., & Shi, J. (2020). Deep learning-based intelligent process monitoring of directed energy deposition in additive manufacturing with thermal images. *Procedia Manufacturing*, 48, 643-649.
- [70]. Li, Y., Hu, Q., Wu, M., Liu, J., & Wu, X. (2016). Extraction and simplification of building façade pieces from mobile laser scanner point clouds for 3D street view services. *ISPRS International Journal of Geo-Information*, 5(12), 231.
- [71]. Li, Y., Ma, L., Zhong, Z., Liu, F., Chapman, M. A., Cao, D., & Li, J. (2020). Deep learning for lidar point clouds in autonomous driving: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8), 3412-3432.
- [72]. Li, Z. (2020). Geospatial big data handling with high performance computing: Current approaches and future directions. *High performance computing for geospatial applications*, 53-76.
- [73]. Li, Z., Hodgson, M. E., & Li, W. (2018). A general-purpose framework for parallel processing of large-scale LiDAR data. *International Journal of Digital Earth*, 11(1), 26-47.
- [74]. Liang, X., Wang, Y., Jaakkola, A., Kukko, A., Kaartinen, H., Hyyppä, J., Honkavaara, E., & Liu, J. (2015). Forest data collection using terrestrial image-based point clouds from a handheld camera compared to terrestrial and personal laser scanning. *IEEE transactions on geoscience and remote sensing*, 53(9), 5117-5132.
- [75]. Lisańczuk, M., Mitelsztadt, K., Parkitna, K., Krok, G., Stereńczak, K., Wysocka-Fijorek, E., & Miścicki, S. (2020). Influence of sampling intensity on performance of two-phase forest inventory using airborne laser scanning. *Forest Ecosystems*, 7(1), 65.
- [76]. Liu, X., Ounifi, H.-A., Gherbi, A., Li, W., & Cheriet, M. (2020). A hybrid GPU-FPGA based design methodology for enhancing machine learning applications performance: X. Liu et al. *Journal of Ambient Intelligence and Humanized Computing*, 11(6), 2309-2323.

- [77]. Long, P., Fan, T., Liao, X., Liu, W., Zhang, H., & Pan, J. (2018). Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning. 2018 IEEE international conference on robotics and automation (ICRA),
- [78]. López-Martínez, F., Núñez-Valdez, E. R., García-Díaz, V., & Bursac, Z. (2020). A case study for a big data and machine learning platform to improve medical decision support in population health management. *Algorithms*, 13(4), 102.
- [79]. Manandhar, A., Zhu, H., Ozkan, E., & Shah, A. (2020). Techno-economic impacts of using a laser-guided variable-rate spraying system to retrofit conventional constant-rate sprayers. *Precision Agriculture*, 21(5), 1156-1171.
- [80]. Marchesini, E., & Farinelli, A. (2020). Discrete deep reinforcement learning for mapless navigation. 2020 IEEE International Conference on Robotics and Automation (ICRA),
- [81]. Marrs, J., & Ni-Meister, W. (2019). Machine learning techniques for tree species classification using co-registered LiDAR and hyperspectral data. *Remote Sensing*, 11(7), 819.
- [82]. Marshall, C., Sturk-Andreaggi, K., Daniels-Higginbotham, J., Oliver, R. S., Barritt-Ross, S., & McMahon, T. P. (2017). Performance evaluation of a mitogenome capture and Illumina sequencing protocol using non-probative, case-type skeletal samples: Implications for the use of a positive control in a next-generation sequencing procedure. *Forensic Science International: Genetics*, 31, 198-206.
- [83]. Mayr, A., Lutz, B., Weigelt, M., Gläsel, T., Kießalt, D., Masuch, M., Riedel, A., & Franke, J. (2018). Evaluation of machine learning for quality monitoring of laser welding using the example of the contacting of hairpin windings. 2018 8th International Electric Drives Production Conference (EDPC),
- [84]. Md Asif Ali Sheak, A., & Risha, A. (2022). The Role of IOT-Driven Energy Analytics in Improving Operational Efficiency and Sustainability in Modern Industrial Facilities. *American Journal of Interdisciplinary Studies*, 3(04), 807-841. <https://doi.org/10.63125/gm63q956>
- [85]. Md. Abdur, R., & Iftekhar, A. (2021). Customer Retention Forecasting in Mobile Wallet Services Using Neural Networks: A Comparative Quantitative Study. *International Journal of Business and Economics Insights*, 1(4), 70-102. <https://doi.org/10.63125/dyrpc387>
- [86]. Mehlretter, M., & Heipke, C. (2019). Cnn-based cost volume analysis as confidence measure for dense matching. 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW),
- [87]. Melia, M. A., Whetten, S. R., Puckett, R., Jones, M., Heiden, M. J., Argibay, N., & Kustas, A. B. (2020). High-throughput additive manufacturing and characterization of refractory high entropy alloys. *Applied Materials Today*, 19, 100560.
- [88]. Meyes, R., Tercan, H., Roggendorf, S., Thiele, T., Büscher, C., Obdenbusch, M., Brecher, C., Jeschke, S., & Meisen, T. (2017). Motion planning for industrial robots using reinforcement learning. *Procedia CIRP*, 63, 107-112.
- [89]. Mohammad Abir Chowdhury, S., & Tonay, P. (2022). An Empirical Analysis of the Impact of Robotic Process Automation and Continuous Auditing on the Timeliness And Accuracy Of Financial Reporting Assurance. *International Journal of Business and Economics Insights*, 2(4), 73-126. <https://doi.org/10.63125/7p8dcr47>
- [90]. Nappi, M., Ricciardi, S., & Tistarelli, M. (2017). Real time 3D face-ear recognition on mobile devices: New scenarios for 3D biometrics “in-the-Wild”. In *Human recognition in unconstrained environments* (pp. 55-75). Elsevier.
- [91]. Nguyen, T. A., Nguyen, P. T., & Do, S. T. (2020). Application of BIM and 3D laser scanning for quantity management in construction projects. *Advances in Civil Engineering*, 2020(1), 8839923.
- [92]. Nino, V., Claudio, D., Valladares, L., & Harris, S. (2020). An enhanced kaizen event in a sterile processing department of a rural hospital: a case study. *International Journal of Environmental Research and Public Health*, 17(23), 8748.
- [93]. Niroui, F., Zhang, K., Kashino, Z., & Nejat, G. (2019). Deep reinforcement learning robot for search and rescue applications: Exploration in unknown cluttered environments. *IEEE Robotics and Automation Letters*, 4(2), 610-617.
- [94]. Nishat, J., & Mst Kaniz, F. (2022). A Review of the Impact of Covid-19 On Maternal and Child Health Service Utilization in Bangladesh. *American Journal of Scholarly Research and Innovation*, 1(02), 223-269. <https://doi.org/10.63125/vgv5s742>
- [95]. O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., & Walsh, J. (2019). Deep learning vs. traditional computer vision. Science and information conference,
- [96]. Okaro, I. A., Jayasinghe, S., Sutcliffe, C., Black, K., Paoletti, P., & Green, P. L. (2019). Automatic fault detection for laser powder-bed fusion using semi-supervised machine learning. *Additive Manufacturing*, 27, 42-53.
- [97]. Omidipoor, M., Toomanian, A., Neysani Samany, N., & Mansourian, A. (2020). Knowledge discovery web service for spatial data infrastructures. *ISPRS International Journal of GeoInformation*, 10(1), 12.
- [98]. Paul, A., Mozaffar, M., Yang, Z., Liao, W.-k., Choudhary, A., Cao, J., & Agrawal, A. (2019). A real-time iterative machine learning approach for temperature profile prediction in additive manufacturing processes. 2019 IEEE international conference on data science and advanced analytics (DSAA),
- [99]. Paulus, S., Dupuis, J., Riedel, S., & Kuhlmann, H. (2014). Automated analysis of barley organs using 3D laser scanning: An approach for high throughput phenotyping. *Sensors*, 14(7), 12670-12686.
- [100]. Pfeiffer, M., Shukla, S., Turchetta, M., Cadena, C., Krause, A., Siegwart, R., & Nieto, J. (2018). Reinforced imitation: Sample efficient deep reinforcement learning for mapless navigation by leveraging prior demonstrations. *IEEE Robotics and Automation Letters*, 3(4), 4423-4430.
- [101]. Pilling, M. J., Henderson, A., & Gardner, P. (2017). Quantum cascade laser spectral histopathology: breast cancer diagnostics using high throughput chemical imaging. *Analytical chemistry*, 89(14), 7348-7355.
- [102]. Poola, D., Salehi, M. A., Ramamohanarao, K., & Buyya, R. (2017). A taxonomy and survey of fault-tolerant workflow management systems in cloud and distributed computing environments. *Software architecture for big data and the cloud*, 285-320.
- [103]. Poorani, A. A., & Sullivan, W. A. (2018). HR analytics: human capital return on investment, productivity, and profit sensitivity: a case of Courtyard Marriott Newark at the University of Delaware. In *Aligning Business Strategies and Analytics: Bridging Between Theory and Practice* (pp. 133-154). Springer.

- [104]. Quentin, D., & Campling, L. (2018). Global inequality chains: Integrating mechanisms of value distribution into analyses of global production. *Global networks*, 18(1), 33-56.
- [105]. Ramus, C., Hovasse, A., Marcellin, M., Hesse, A.-M., Mouton-Barbosa, E., Bouyssié, D., Vaca, S., Carapito, C., Chaoui, K., & Bruley, C. (2016). Benchmarking quantitative label-free LC-MS data processing workflows using a complex spiked proteomic standard dataset. *Journal of proteomics*, 132, 51-62.
- [106]. Rathinaraja, J., Ananthanarayana, V., & Paul, A. (2019). Dynamic ranking-based MapReduce job scheduler to exploit heterogeneous performance in a virtualized environment: J. Rathinaraja et al. *The Journal of Supercomputing*, 75(11), 7520-7549.
- [107]. Ravi, R., Lin, Y.-J., Shamseldin, T., Elbahnasawy, M., Crawford, M., & Habib, A. (2018). Implementation of uav-based lidar for high throughput phenotyping. IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium,
- [108]. Richter, R., & Döllner, J. (2014). Concepts and techniques for integration, analysis and visualization of massive 3D point clouds. *Computers, Environment and Urban Systems*, 45, 114-124.
- [109]. Rishav, R., Battrawy, R., Schuster, R., Wasenmüller, O., & Stricker, D. (2020). Deeplidarflow: A deep learning architecture for scene flow estimation using monocular camera and sparse lidar. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),
- [110]. Sampedro, C., Bavle, H., Rodriguez-Ramos, A., de La Puente, P., & Campoy, P. (2018). Laser-based reactive navigation for multirotor aerial robots using deep reinforcement learning. 2018 IEEE/RSJ international Conference on intelligent Robots and Systems (IROS),
- [111]. Seel, J., Barbose, G. L., & Wiser, R. H. (2014). An analysis of residential PV system price differences between the United States and Germany. *Energy Policy*, 69, 216-226.
- [112]. Shafiekhani, A., Kadam, S., Fritschi, F. B., & DeSouza, G. N. (2017). Vinobot and vinocular: Two robotic platforms for high-throughput field phenotyping. *Sensors*, 17(1), 214.
- [113]. Shaiful, M., Anisur, R., & Md, A. (2022). A Systematic Literature Review on the Role of Digital Health Twins in Preventive Healthcare For Personal and Corporate Wellbeing. *American Journal of Interdisciplinary Studies*, 3(04), 1-31. <https://doi.org/10.63125/negjw373>
- [114]. Shao, S., Mallery, K., & Hong, J. (2020). Machine learning holography for measuring 3D particle distribution. *Chemical Engineering Science*, 225, 115830.
- [115]. Shevchik, S., Le-Quang, T., Meylan, B., Farahani, F. V., Olbinado, M. P., Rack, A., Masinelli, G., Leinenbach, C., & Wasmer, K. (2020). Supervised deep learning for real-time quality monitoring of laser welding with X-ray radiographic guidance. *Scientific reports*, 10(1), 3389.
- [116]. Shi, H., Shi, L., Xu, M., & Hwang, K.-S. (2019). End-to-end navigation strategy with deep reinforcement learning for mobile robots. *IEEE Transactions on Industrial Informatics*, 16(4), 2393-2402.
- [117]. Shirowzhan, S., Sepasgozar, S. M., Li, H., Trinder, J., & Tang, P. (2019). Comparative analysis of machine learning and point-based algorithms for detecting 3D changes in buildings over time using bi-temporal lidar data. *Automation in construction*, 105, 102841.
- [118]. Sladić, D., Radulović, A., & Govedarica, M. (2020). Development of process model for Serbian cadastre. *Land Use Policy*, 98, 104273.
- [119]. Soroush, M., Mehrtash, A., Khazraee, E., & Ur, J. A. (2020). Deep learning in archaeological remote sensing: Automated qanat detection in the Kurdistan region of Iraq. *Remote Sensing*, 12(3), 500.
- [120]. Sözer, H. (2020). Humanitarianism with a neo-liberal face: Vulnerability intervention as vulnerability redistribution. *Journal of Ethnic and Migration Studies*, 46(11), 2163-2180.
- [121]. Sundaramurthy, S. C., Wesch, M., Ou, X., McHugh, J., Rajagopalan, S. R., & Bardas, A. (2017). Humans are dynamic. our tools should be too. innovations from the anthropological study of security operations centers. *IEEE Internet Computing*.
- [122]. Tang, W., & Feng, W. (2017). Parallel map projection of vector-based big spatial data: Coupling cloud computing with graphics processing units. *Computers, Environment and Urban Systems*, 61, 187-197.
- [123]. Taş, Ö. Ş., Kuhnt, F., Zöllner, J. M., & Stiller, C. (2016). Functional system architectures towards fully automated driving. 2016 IEEE Intelligent vehicles symposium (IV),
- [124]. Thapa, S., Zhu, F., Walia, H., Yu, H., & Ge, Y. (2018). A novel LiDAR-based instrument for high-throughput, 3D measurement of morphological traits in maize and sorghum. *Sensors*, 18(4), 1187.
- [125]. Trivedi, H., Mesterhazy, J., Laguna, B., Vu, T., & Sohn, J. H. (2018). Automatic determination of the need for intravenous contrast in musculoskeletal MRI examinations using IBM Watson's natural language processing algorithm. *Journal of digital imaging*, 31(2), 245-251.
- [126]. Tu, X., Xu, C., Liu, S., Lin, S., Chen, L., Xie, G., & Li, R. (2020). LiDAR point cloud recognition and visualization with deep learning for overhead contact inspection. *Sensors*, 20(21), 6387.
- [127]. Valero, E., Forster, A., Bosché, F., Hyslop, E., Wilson, L., & Turmel, A. (2019). Automated defect detection and classification in ashlar masonry walls using machine learning. *Automation in construction*, 106, 102846.
- [128]. Veselkov, K., Sleeman, J., Claude, E., Vissers, J. P., Galea, D., Mroz, A., Laponogov, I., Towers, M., Tonge, R., & Mirnezami, R. (2018). BASIS: High-performance bioinformatics platform for processing of large-scale mass spectrometry imaging data in chemically augmented histology. *Scientific reports*, 8(1), 4053.
- [129]. Vo, A. V., Laefer, D. F., Smolic, A., & Zolanvari, S. I. (2019). Per-point processing for detailed urban solar estimation with aerial laser scanning and distributed computing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 155, 119-135.

- [130]. Wahab, H., Jain, V., Tyrrell, A. S., Seas, M. A., Kotthoff, L., & Johnson, P. A. (2020). Machine-learning-assisted fabrication: Bayesian optimization of laser-induced graphene patterning using in-situ Raman analysis. *Carbon*, 167, 609-619.
- [131]. Wang, H., Guan, X., & Wu, H. (2017). A hybrid parallel spatial interpolation algorithm for massive LiDAR point clouds on heterogeneous CPU-GPU systems. *ISPRS International Journal of Geo-Information*, 6(11), 363.
- [132]. Wang, M., Cui, Y., Wang, X., Xiao, S., & Jiang, J. (2017). Machine learning for networking: Workflow, advances and opportunities. *Ieee Network*, 32(2), 92-99.
- [133]. Wang, S., Zhong, Y., & Wang, E. (2019). An integrated GIS platform architecture for spatiotemporal big data. *Future Generation Computer Systems*, 94, 160-172.
- [134]. Wang, T., Zhang, W., Xu, J., & Gu, Z. (2020). Workflow-aware automatic fault diagnosis for microservice-based applications with statistics. *IEEE Transactions on Network and Service Management*, 17(4), 2350-2363.
- [135]. Wu, C.-J., Brooks, D., Chen, K., Chen, D., Choudhury, S., Dukhan, M., Hazelwood, K., Isaac, E., Jia, Y., & Jia, B. (2019). Machine learning at facebook: Understanding inference at the edge. 2019 IEEE international symposium on high performance computer architecture (HPCA),
- [136]. Wu, C. Q., Lin, X., Yu, D., Xu, W., & Li, L. (2014). End-to-end delay minimization for scientific workflows in clouds under budget constraint. *IEEE Transactions on Cloud Computing*, 3(2), 169-181.
- [137]. Wu, Y., Wang, Y., Zhang, S., & Ogai, H. (2020). Deep 3D object detection networks using LiDAR data: A review. *IEEE Sensors Journal*, 21(2), 1152-1171.
- [138]. Xiang, J., Li, Q., Dong, X., & Ren, Z. (2019). Continuous control with deep reinforcement learning for mobile robot navigation. 2019 Chinese Automation Congress (CAC),
- [139]. Xie, G., Wei, Y., Le, Y., & Li, R. (2019). Redundancy minimization and cost reduction for workflows with reliability requirements in cloud-based services. *IEEE Transactions on Cloud Computing*, 10(1), 633-647.
- [140]. Xie, X., Zhao, M., He, J., & Zhou, B. (2019). Automatic and visual processing method of non-contact monitoring for circular stormwater sewage tunnels based on LiDAR data. *Energies*, 12(9), 1599.
- [141]. Xie, Y., Tian, J., & Zhu, X. X. (2020). Linking points with labels in 3D: A review of point cloud semantic segmentation. *IEEE Geoscience and remote sensing magazine*, 8(4), 38-59.
- [142]. Yang, Y., Li, M., & Ma, X. (2020). A Point Cloud Simplification Method Based on Modified Fuzzy C-Means Clustering Algorithm with Feature Information Reserved. *Mathematical Problems in Engineering*, 2020(1), 5713137.
- [143]. Yang, Y., Pan, L., Ma, J., Yang, R., Zhu, Y., Yang, Y., & Zhang, L. (2020). A high-performance deep learning algorithm for the automated optical inspection of laser welding. *Applied Sciences*, 10(3), 933.
- [144]. Yang, Z., Nguyen, P., Jin, H., & Nahrstedt, K. (2019). MIRAS: Model-based reinforcement learning for microservice resource allocation over scientific workflows. 2019 IEEE 39th international conference on distributed computing systems (ICDCS),
- [145]. Yu, Y., Li, J., Guan, H., Jia, F., & Wang, C. (2014). Learning hierarchical features for automated extraction of road markings from 3-D mobile LiDAR point clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(2), 709-726.
- [146]. Yue, P., Gao, F., Shangguan, B., & Yan, Z. (2020). A machine learning approach for predicting computational intensity and domain decomposition in parallel geoprocessing. *International Journal of Geographical Information Science*, 34(11), 2243-2274.
- [147]. Zhang, C., Patras, P., & Haddadi, H. (2019). Deep learning in mobile and wireless networking: A survey. *IEEE Communications surveys & tutorials*, 21(3), 2224-2287.
- [148]. Zhang, H., Zhang, G., & Yan, Q. (2019). Digital twin-driven cyber-physical production system towards smart shop-floor. *Journal of Ambient Intelligence and Humanized Computing*, 10(11), 4439-4453.
- [149]. Zhang, J., Xu, L., Zhang, Y., Liu, G., Zhao, L., & Wang, Y. (2019). An on-demand scalable model for geographic information system (GIS) data processing in a cloud GIS. *ISPRS International Journal of Geo-Information*, 8(9), 392.
- [150]. Zhang, T., & Li, J. (2015). Online task scheduling for LiDAR data preprocessing on hybrid GPU/CPU devices: A reinforcement learning approach. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(1), 386-397.
- [151]. Zhang, T., & Mao, S. (2020). Machine learning for end-to-end congestion control. *IEEE Communications Magazine*, 58(6), 52-57.
- [152]. Zhou, J., Tan, X., Shao, Z., & Ma, L. (2019). FVNet: 3D front-view proposal generation for real-time object detection from point clouds. 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI),
- [153]. Zhu, J., & Pierskalla Jr, W. P. (2016). Applying a weighted random forests method to extract karst sinkholes from LiDAR data. *Journal of Hydrology*, 533, 343-352.