Review

Not peer-reviewed version

Synchronization, Optimization, and Adaptation of Machine Learning Techniques for Computer Vision in Cyber-Physical Systems: A Comprehensive Analysis

Kai Hung Tang , Mohamed Chahine Ghanem * , Vassil Vassilev , Karim Ouazzane , Pawel Gasiorowski

Posted Date: 10 February 2025

doi: 10.20944/preprints202501.0521.v2

Keywords: Machine Learning; Adaptation; Synchronization; Optimization; Computer vision; Cyber-physical systems

 doi:10.20944/preprints202501.0521.v2

Article

Synchronization, Optimization, and Adaptation of Machine Learning Techniques for Computer Vision in Cyber-Physical Systems: A Comprehensive Analysis

Kai Hung Tang [1], Mohamed Chahine Ghanem [1,2,*], Vassil Vassilev [1], Karim Ouazzane [1] and Pawel Gasiorowski [1]

[1] Cyber Security Research Centre, London Metropolitan University, London, UK
[2] Department of Computer Science, University of Liverpool, UK
* Correspondence: ghanemm@staff.londonmet.ac.uk

Abstract: Cyber-Physical Systems (CPS) seamlessly integrate computers, networks, and physical devices, enabling machines to communicate, process data, and respond to real-world conditions in real-time. By bridging the digital and physical worlds, CPS ensures operations that are efficient, safe, innovative, and controllable. As smart cities and autonomous machines become more prevalent, understanding CPS is crucial for driving future progress. Recent advancements in edge computing, AI-driven vision, and collaborative systems have significantly enhanced CPS capabilities. synchronization, optimization, and adaptation are intricate processes that impact CPS performance across different domains. Therefore, identifying emerging trends and uncovering research gaps is essential to highlight areas that require further investigation and improvement. This systematic review and analysis aims to offer a unique point to researcher and facilitates this process by allowing researchers to benchmark and compare various techniques, evaluate their effectiveness, and establish best practices. It provides evidence-based insights into optimal strategies for implementation while addressing potential trade-offs in performance, resource usage, and reliability. Additionally, such reviews help identify widely accepted standards and frameworks, contributing to the development of standardized approaches.

Keywords: machine learning algorithm; computer vision; cyber-physical systems

1. Introduction

1.1. Context and Importance

This paper focuses on the integration of machine learning (ML) techniques with computer vision (CV) to address the evolving demands of cyber-physical systems (CPS). CPS, which combines computational and physical processes, increasingly relies on CV for real-time perception and decision-making. These systems span various applications, including autonomous vehicles, smart grids, industrial automation, healthcare devices, and intelligent transportation networks. The real-time capabilities provided by CV enable CPS to interpret complex visual data from their environment, facilitating tasks such as object detection, scene understanding, and adaptive control.

However, synchronizing and optimizing ML models for such applications remains a critical challenge, given CPS's dynamic and resource-constrained nature. Key issues include ensuring low-latency processing, maintaining accuracy under varying operational conditions, and efficiently managing computational resources, particularly in embedded or edge-computing scenarios. Furthermore, CPS often operates in unpredictable and sometimes harsh environments, requiring robust ML models that can handle noisy or incomplete data without compromising performance.

Another dimension of the challenge involves the continuous adaptation of ML algorithms to evolving data patterns and system behaviours. CPS needs adaptive learning strategies to update models in real-time or near-real-time. This demands advanced techniques such as incremental learning,

 doi:10.20944/preprints202501.0521.v2

transfer learning, and federated learning, which allow models to evolve based on new information without the need for complete retraining from scratch.

This paper explores these multifaceted challenges, reviewing recent advancements and identifying key areas for future research. By addressing these issues, we aim to pave the way for more efficient, reliable, and adaptable ML-integrated CV solutions in next-generation CPSs.

1.2. Problem Statement

Despite advancements in ML and CV, their deployment in CPS faces several challenges:

1. Synchronization issues due to heterogeneous hardware and real-time constraints. CPS environments often consist of diverse hardware components. Ensuring seamless integration and real-time data processing across these heterogeneous platforms is complex. Synchronization becomes particularly challenging when multiple sensors and processing units work together to provide a coherent and timely response. Variations in processing power, data transfer rates, and latency can lead to discrepancies or delays undermining the system's overall performance. Addressing these issues requires sophisticated algorithms and synchronization protocols that can harmonize the operation of different hardware components while meeting stringent real-time constraints.
2. Optimization difficulties related to balancing accuracy and computational efficiency. ML models, particularly deep learning architectures, often demand substantial computational resources to achieve high accuracy. In CPS, where real-time decision-making is crucial, striking a balance between model performance and computational efficiency is essential. Resource-constrained environments, such as embedded systems or edge devices, may not have the capacity to run large models or handle intensive computations. Therefore, optimizing models to deliver accurate predictions without overloading system resources is a significant challenge. Techniques such as model pruning, quantization, and knowledge distillation are commonly explored, but implementing them effectively without compromising performance remains an ongoing area of research.
3. Adaptation requirements to ensure robust performance across varying environments and tasks. CPS often operates in dynamic and unpredictable environments where conditions can change rapidly. For instance, an autonomous vehicle must adapt to different weather conditions, lighting variations, and traffic scenarios. Similarly, industrial CPS must handle fluctuations in sensor data and operational conditions. ML models trained in controlled settings may struggle to maintain accuracy when faced with such variability. This necessitates adaptive learning strategies and robust models capable of generalising across different tasks and environments. Techniques such as transfer learning, online learning, and domain adaptation are crucial, but integrating them into CPS without causing disruptions or requiring constant retraining poses significant challenges.

Addressing these challenges is essential for the effective deployment of ML for CV in CPS, ensuring these systems can operate reliably, efficiently, and safely in real-world applications. This paper explores potential solutions and innovations aimed at overcoming these hurdles, paving the way for more resilient and adaptable CPS architectures.

1.3. Objectives

We aim to synthesize existing research on ML techniques for CV in CPS. This involves examining a wide range of methodologies, including traditional approaches, advanced deep learning architectures, and other methods, to understand their applications, strengths, and limitations. The review will cover various CV tasks relevant to CPS, such as object detection, image classification, semantic segmentation, and anomaly detection. By analyzing existing literature, we intend to highlight the most effective strategies, key milestones, and technological advancements that have shaped this interdisciplinary field. This synthesis will serve as a foundation for understanding how ML-driven CV solutions contribute to enhancing the functionality and reliability of CPS across different domains, including autonomous

 doi:10.20944/preprints202501.0521.v2

vehicles, smart manufacturing, and healthcare systems. Despite significant progress, this review seeks to identify and analyze existing gaps related to synchronization, optimization, and adaptation. In terms of synchronization, we will examine the complexities of integrating heterogeneous hardware components and maintaining real-time performance across diverse CPS platforms. For optimization, we will explore the trade-offs between computational efficiency and model accuracy, particularly in resource-constrained environments. Regarding adaptation, we aim to uncover the limitations of current ML models in handling dynamic and unpredictable environments, where robust performance is essential. By systematically identifying these gaps, we hope to provide a clearer picture of the unresolved issues that need to be addressed to enable more effective and reliable ML-CV integration in CPS.

This review will propose future directions for research and development in this interdisciplinary domain. The recommendations will focus on key areas such as developing more efficient and adaptable algorithms, enhancing real-time synchronization frameworks, and designing robust models capable of operating under varying conditions. Additionally, we will highlight the importance of interdisciplinary collaboration and domain-specific experts in addressing these complex challenges holistically. Emerging trends, such as edge computing, federated learning, and hybrid models combining symbolic reasoning with neural networks, will also be discussed as potential avenues for innovation. By outlining these future directions, we aim to inspire further research and development efforts, ultimately contributing to the evolution of smarter, more efficient, and resilient CPSs.

1.4. Structure

The remainder of this article is organized as follows: Chapter 2 outlines the systematic review process. Chapter 3 provides foundational knowledge on ML, CV, and CPS. Chapter 4 synthesizes key findings and identifies emerging themes. Chapter 5 evaluates current research and explores future opportunities. Finally, Chapter 6 offers practical insights and Chapter 7 summarises the significance of the study. The structure of this article is illustrated in Figure 1.

 doi:10.20944/preprints202501.0521.v2

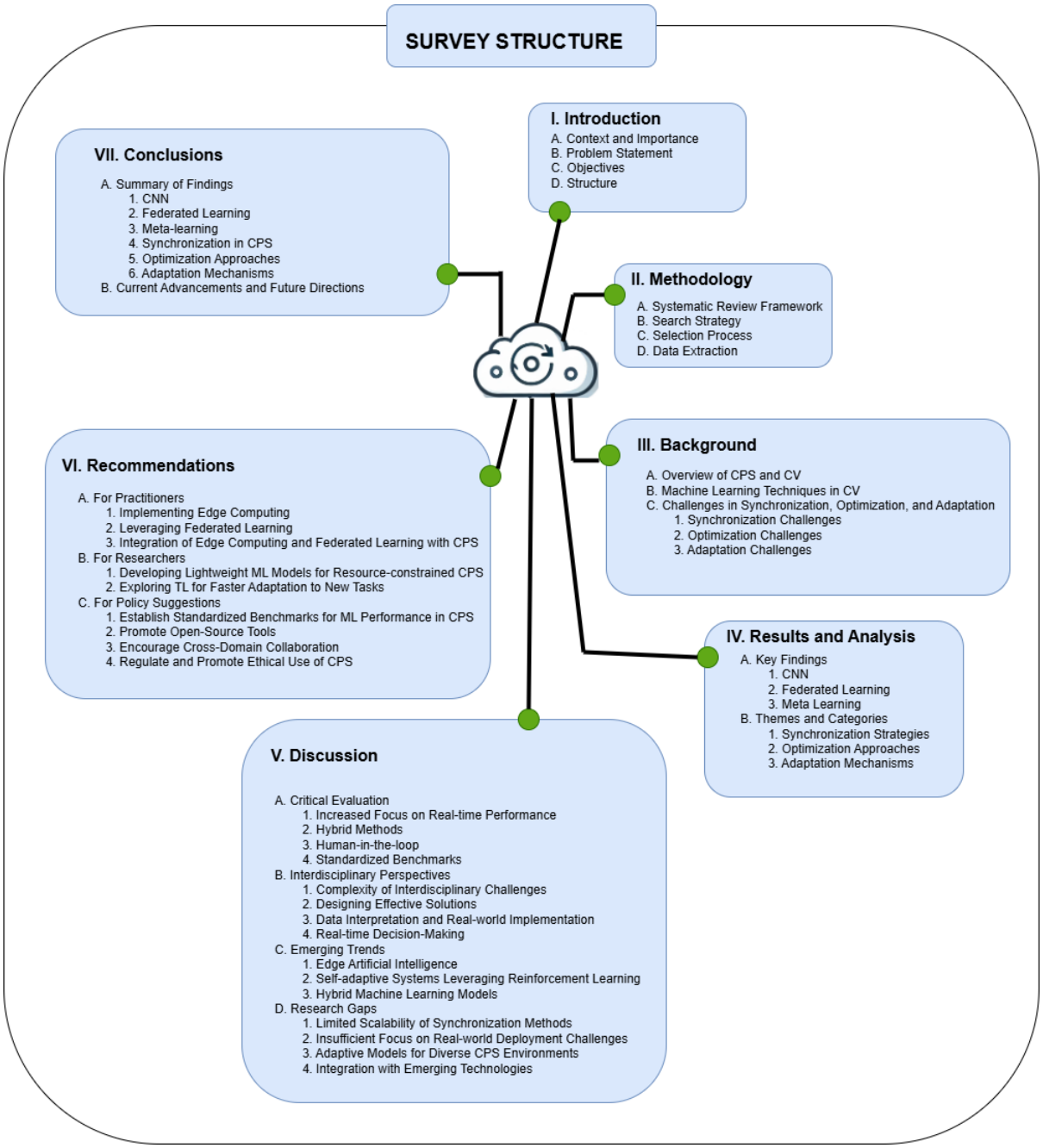

Figure 1. Structure of the Article

2. Methodology

2.1. Systematic Review Framework

The review follows a systematic framework, adhering to the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) [1] guidelines, ensuring a thorough and transparent evaluation of the relevant literature. The PRISMA framework involves several critical steps, including developing a detailed research protocol, conducting comprehensive and reproducible literature searches across multiple databases, and applying predefined inclusion and exclusion criteria for study selection.

By adhering to these guidelines, the review minimises bias and enhances reliability. The methodology involves a two-phase screening process (title/abstract and full-text reviews) conducted by independent reviewers, and discrepancies resolved by consensus. Data extraction is performed using standardised forms to capture key study characteristics, findings, and quality assessments. In addition,

 doi:10.20944/preprints202501.0521.v2

a PRISMA flow diagram is presented to visually illustrate the search process, the number of studies identified, screened, and included, as well as the reasons for exclusions.

This systematic approach ensures comprehensive coverage of the literature and facilitates transparency and replicability, enabling other researchers to validate and build upon the findings.

2.2. Search Strategy

When conducting academic research, we have used multiple scholarly databases can ensure comprehensive coverage of relevant literature. Databases like IEEE Xplore, SpringerLink, Scopus, and Google Scholar provide unique advantages for finding peer-reviewed articles and conference papers.

For the search process, the following keywords have been used: Machine Learning, Computer Vision in Cyber-Physical Systems, synchronization in Machine Learning, and optimization and Adaptation of Computer Vision Algorithms. Start by entering keywords into title/abstract and then into full-text reviewers, we combine the keywords with AND or OR to explore related works and access citations.

To consider the most recent works in the field, the search period is limited between 2010 and 2024. However, in some cases, it was necessary to use older preliminary references to get an overview of all the basic notions and fully cover the study's topic. Only papers on ML for CV, emphasizing studies addressing synchronization, optimization, or adaptation in CPS have been considered. Inclusion criteria focused on peer-reviewed publications from 2010 to 2023, emphasizing studies addressing synchronization, optimization, or adaptation in CPS.

2.3. Selection Process

Initially, keywords were entered into the title and abstract search fields to identify articles directly addressing the core research topics. Following this preliminary screening, full-text reviews were conducted to assess the relevance and depth of the selected works concerning our research objectives. Boolean operators such as AND and OR were used to combine these keywords, allowing us to refine searches, link interconnected concepts, and identify relevant citations more effectively. By strategically utilising comprehensive databases and systematically enhancing search methodologies, we aimed to construct a robust overview of the current research landscape, highlighting existing gaps and opportunities for future exploration [2,3] and [4].

To assess the quality and relevance of the studies, we utilized established metrics such as citation impact and methodological rigour. Additionally, we assigned a qualitative score ranging from 0 to 5 to evaluate how effectively each study addressed our research questions. A score of 5 indicated a strong alignment between the study's research question and ours, without suggesting duplication. This scoring system provided a structured framework for systematically evaluating the relevance and comprehensiveness of each study within the context of our research objectives [5,6].

The temporal distribution of the selected articles, shown in Figure 2, reveals a notable upward trend, with a significant surge in publications over the last two years. This trend underscores the growing interest and rapid acceleration in research focusing on ML algorithms for CV applications within CPS.

 doi:10.20944/preprints202501.0521.v2

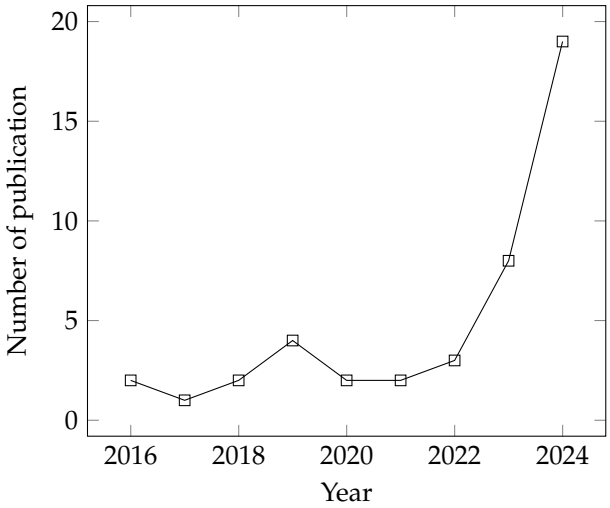

Figure 2. Distribution of the publications between 2016 and 2024

2.4. Data Extraction

Data extraction involved identifying and recording key data points critical to our studies for CV applications in CPS. The first category of data focused on ML models and architectures utilized, including specific algorithms, frameworks, and design patterns employed in the selected articles. This information was vital for understanding the underlying computational approaches and their suitability for CPS applications. Another important area of focus was the synchronization strategies between ML algorithms and CPS hardware. This encompassed methods used to ensure smooth integration and coordination between the computational components of ML systems and the physical processes controlled by CPS. Details included timing mechanisms, communication protocols, and any co-design considerations.

We also extracted information on optimization techniques for resource-constrained environments, emphasizing strategies used to adapt ML operations for hardware with limited computational power, energy, or memory. These data points provided insights into practical implementations where resource efficiency was a critical constraint.

Lastly, we gathered data on adaptation methods for dynamic operational contexts, which included techniques used to modify or retrain ML models in response to changing environmental conditions or system demands. This category highlighted how studies addressed the challenges of real-time adaptability and resilience in CPS applications. The search article flow chart is shown in Figure 3

Collectively, these data points formed a comprehensive basis to analyze trends, innovations, and gaps in CV application to CPS, allowing for a robust evaluation of current methodologies and their implications.

 doi:10.20944/preprints202501.0521.v2

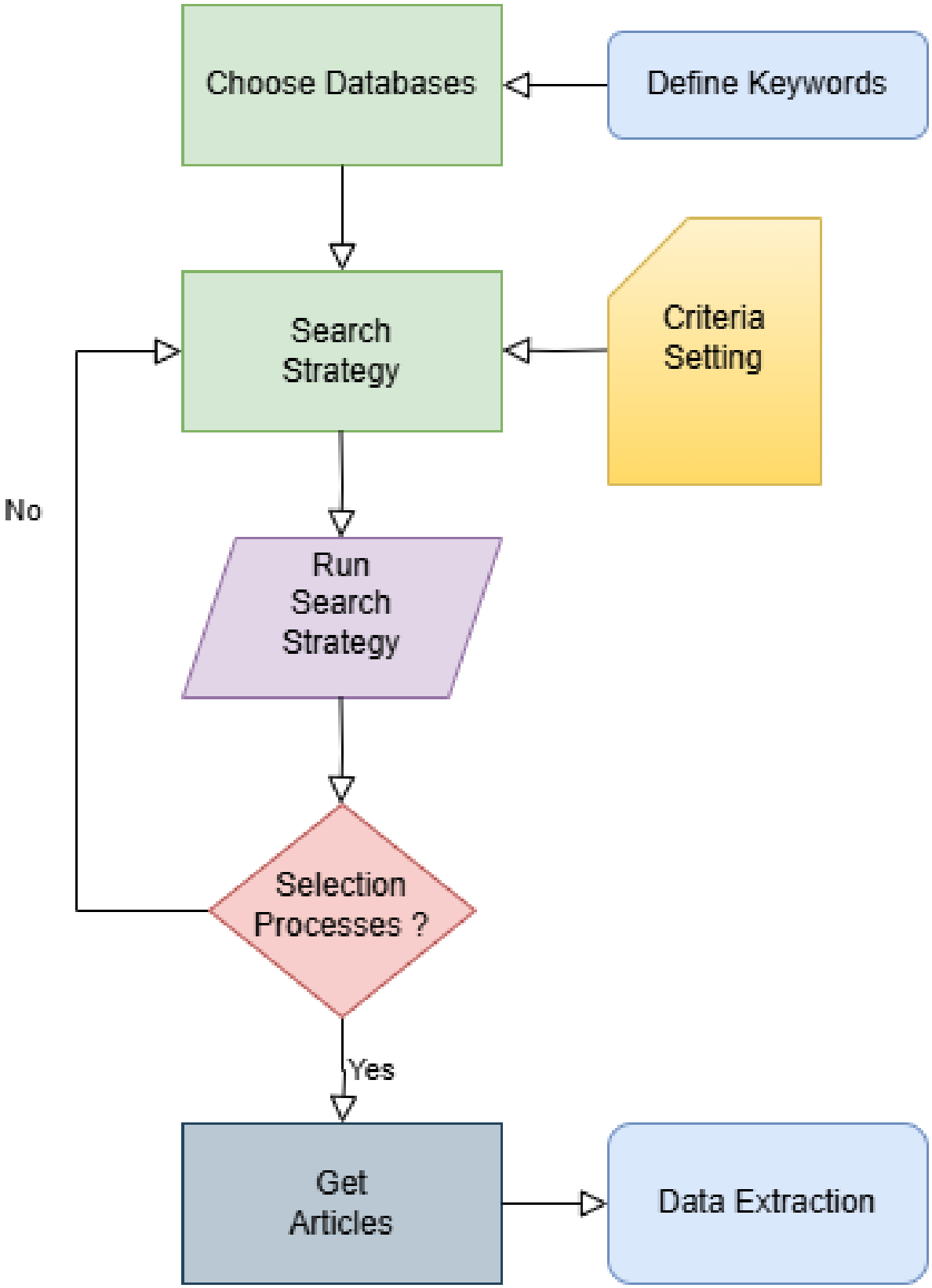

Figure 3. Papers Search and Review Flowchart

3. Background

3.1. Overview of Cyber-Physical Systems (CPS) and Computer Vision (CV)

CPSs integrate computing elements with physical processes to enable real-time monitoring and control. These systems bridge the physical and digital worlds, driving advancements in smart grids,

 doi:10.20944/preprints202501.0521.v2

autonomous vehicles, industrial automation, and healthcare. Figure 4 illustrates CPS application domains.

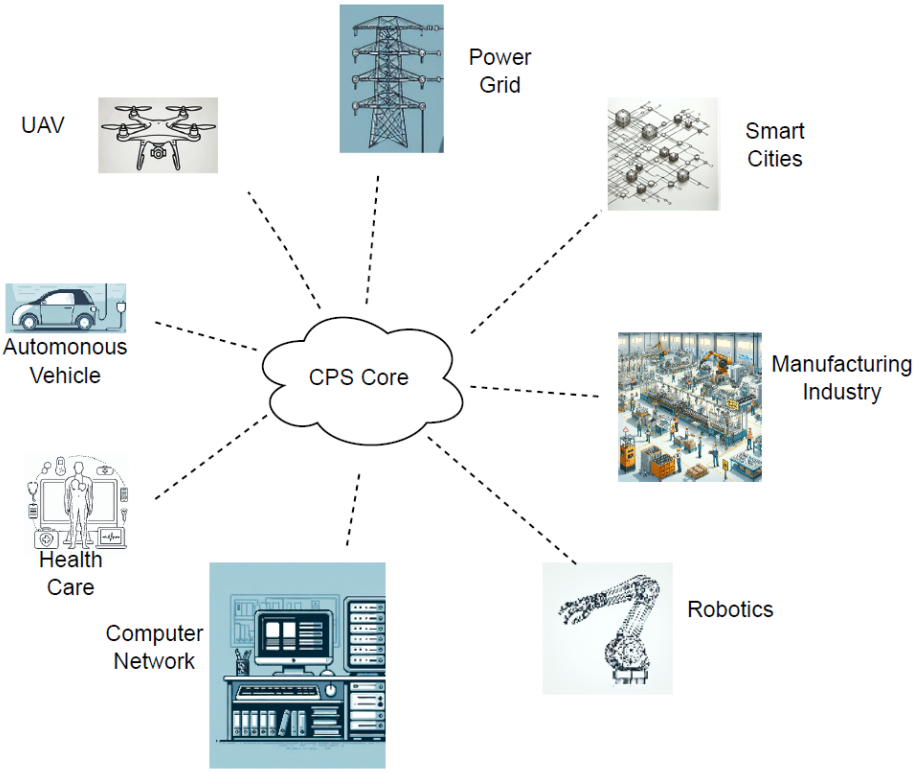

Figure 4. CPS Domaines and Applications

The Core components of CPS include:

- Sensors: collect data from the physical environment, converting real-world information into digital signals. Examples include temperature sensors, cameras, LIDAR, GPS, and accelerometers. In CPS, sensors play a vital role in:
  - Monitoring environmental conditions (e.g., in smart buildings).
  - Detecting anomalies in industrial processes.
  - Providing input for control decisions in autonomous vehicles.
- Actuators: perform actions based on decisions made by the computational units, transforming digital commands into physical actions. They can control various devices, such as motors, valves, or robotic arms. Key functions include:
  - Adjusting machinery operations in manufacturing.
  - Steering autonomous vehicles based on sensor data.
  - Regulating power distribution in smart grids.
- Computational Units: process sensor data, run control algorithms, and send commands to actuators. They can range from embedded microcontrollers to powerful cloud-based systems. Functions include:
  - Real-time data analysis.
  - Running predictive models to anticipate system behaviours.
  - Ensuring system security and reliability through robust software protocols.

Recent advancements in edge computing, AI-driven vision, and collaborative systems continue to extend CPS capabilities. The functioning of CPS is grounded in real-time data from the physical environment to guide decision-making and actions. CV enhances CPS in the ways below:

 doi:10.20944/preprints202501.0521.v2

3.1.1. Perception and Sensing

CV acts as the "eyes" of CPS, gathering visual data through cameras and sensors. It is critical for autonomous vehicles, drones, and industrial robots, where vision algorithms extract features for object recognition, motion detection, depth estimation, and tracking to have real-time scene understanding. An example application in autonomous vehicles is that CV detects pedestrians, vehicles, traffic signals, and road conditions to provide inputs for the control system.

3.1.2. Real-Time Monitoring and Feedback

CPS relies on real-time feedback from the physical environment to function efficiently. Computer vision (CV) facilitates this by capturing and interpreting visual data in real-time, enabling systems to dynamically adjust their actions or decisions based on changes in their surroundings. One of the defining features of CPS is its real-time operation. In industrial environments, robots continuously monitor production, identify flaws, optimize performance, and anticipate potential issues to prevent malfunctions. This minimizes human intervention while enhancing production speed. Similarly, CPS-enabled drones navigate and avoid obstacles during deliveries, while smart home systems automatically adjust lighting and temperature in response to current conditions.

3.1.3. Autonomy and Decision Making

CPSs harness CV, AI, and ML to enable autonomous decision-making. Vision systems analyze large volumes of visual data, identify patterns, and make independent decisions without human intervention. For example, drones use vision-based navigation to autonomously avoid obstacles and inspect critical infrastructure such as bridges and power lines. In healthcare, CPS supports continuous patient monitoring through wearable devices, robotic surgical tools, and advanced prosthetics, delivering accurate and timely medical care. Similarly, smart grids optimize energy efficiency by monitoring consumption and distributing power more effectively, reducing waste and improving overall resource management.

3.1.4. Safety and Surveillance

CV plays a vital role in safety and security, particularly in smart cities and industries. Vision-based systems can detect objects, identify faces or license plates, and trigger alarms in response to suspicious activity. Vision-enabled surveillance systems in smart grids monitor critical infrastructure for breaches or abnormalities caused by intrusions or equipment malfunctions. Self-driving cars also depend on CPS to process vast amounts of data in real-time, enabling split-second decisions that enhance road safety.

3.1.5. Human-Machine Interfaces

CV enables human-machine interfaces by interpreting human gestures, motions, or expressions, allowing systems to interact with humans in real-time. It is widely used in smart gadgets, healthcare, and robotics. In healthcare, vision systems track patient movements or facial expressions to monitor health conditions or assist in physical therapy.

3.2. Machine Learning Techniques in CV

The following section explores commonly used ML models in CV, highlighting their architectures, functionalities, and applications:

3.2.1. Convolutional Neural Networks (CNNs) for Image Recognition

CNNs are cornerstones in computer vision, designed specifically to handle grid-like data such as images. Inspired by the human visual cortex, CNNs use a series of convolutional layers to automatically and adaptively learn spatial hierarchies of features from input images. These networks apply filters (kernels) that slide over the image, detecting patterns such as edges, textures, and complex objects at different layers. CNNs are widely used for tasks such as image classification, object detection, and

 doi:10.20944/preprints202501.0521.v2

semantic segmentation. A schematic diagram of a basic CNN architecture is shown in Figure 5 and key components of CNNs include:

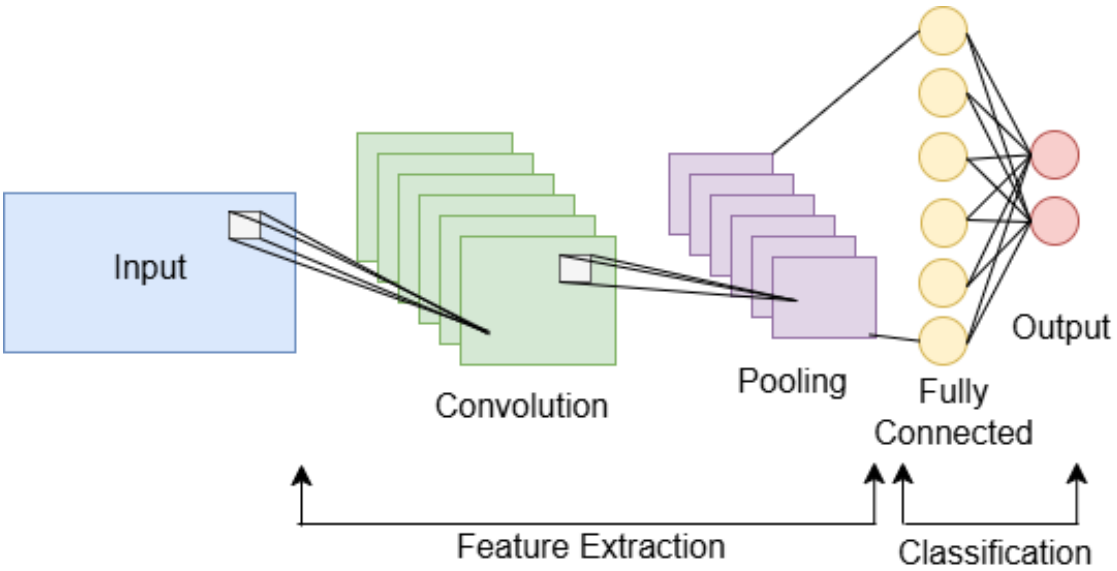

Figure 5. Schematic diagram of a basic CNN architecture [7]

- Convolutional Layers: These layers apply convolution operations to the input image, using filters (or kernels) to detect various features such as edges, textures, and patterns.
- Activation Functions: After convolution, activation functions are applied to introduce non-linearity, helping the network learn more complex patterns.
- Pooling Layers: These layers reduce the dimensionality of feature maps, preserving essential information while minimizing computational load and making the network more robust to variations in input.
- Fully Connected Layers: After several convolutional and pooling layers, these layers combine the features to make predictions or classifications.
- Output Layer: The final layer usually uses a softmax activation function to produce a probability distribution over the possible classes, allowing the network to make a prediction.

3.2.2. Recurrent Neural Networks (RNNs) for Sequential Data Processing

While primarily designed for sequential or time-series data, RNNs have found applications in computer vision, particularly in tasks involving sequences of images or video data. RNNs are unique in their ability to process sequences by maintaining a hidden state that captures information about previous elements in the sequence. This makes them effective for modeling temporal dependencies, making them useful for tasks like stock price prediction and weathering forecasting. By analyzing frames sequentially, RNNs can be used for tasks like action recognition in videos. For instance, they can process sequences of video frames to recognize activities (e.g., walking, running) or generate textual descriptions for images.

Figure 6 illustrates the variants of RNNs [8], accompanied by their descriptions below:

- Long Short-Term Memory (LSTM): are a type of RNN designed to address the vanishing gradient problem, allowing them to capture long-term dependencies more effectively.
- Gated Recurrent Unit (GRU): are a simplified version of LSTMs that also help mitigate the vanishing gradient problem while being computationally more efficient.

 doi:10.20944/preprints202501.0521.v2

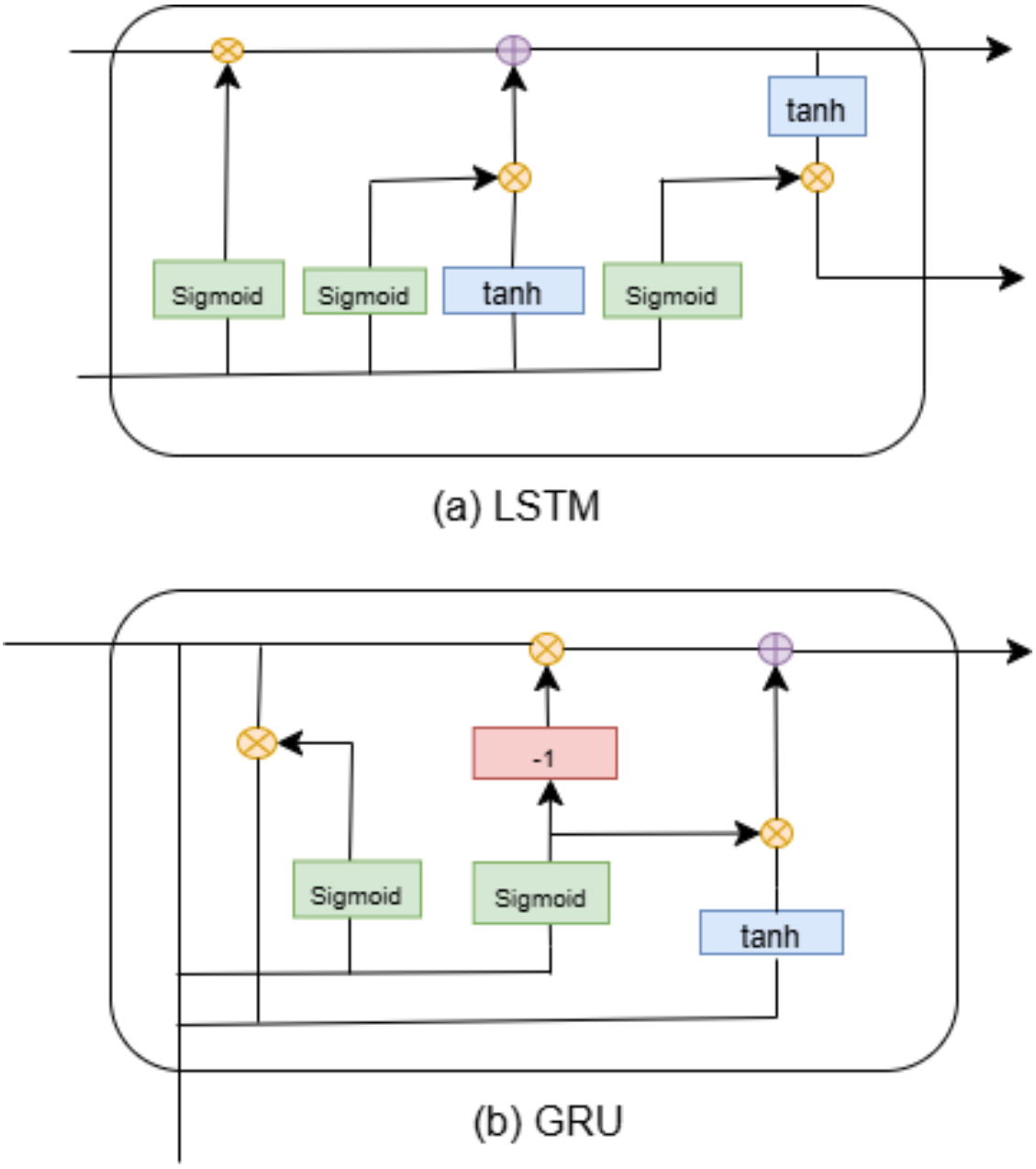

Figure 6. Basic structure of the a) LSTM and b) GRU neural networks. [8]

3.2.3. Transformer-Based Architectures for Advanced Feature Extraction

Transformers, initially developed for natural language processing (NLP), have transformed deep learning with their attention mechanisms, enabling models to capture global relationships within input sequences. In computer vision, architectures such as the Vision Transformer (ViT) apply these principles to image data, offering robust feature extraction and representation capabilities. A view of the model is shown in Figure 7 [9]. Transformer-based models perform exceptionally well in tasks like image classification, object detection, and segmentation. Their core concepts include the self-attention mechanism and patch embedding. The self-attention mechanism allows models to assess the importance of different image regions, capturing long-range dependencies and contextual relationships. Patch embedding converts an image into a sequence of fixed-size patches, analogous to word tokens in NLP.

 doi:10.20944/preprints202501.0521.v2

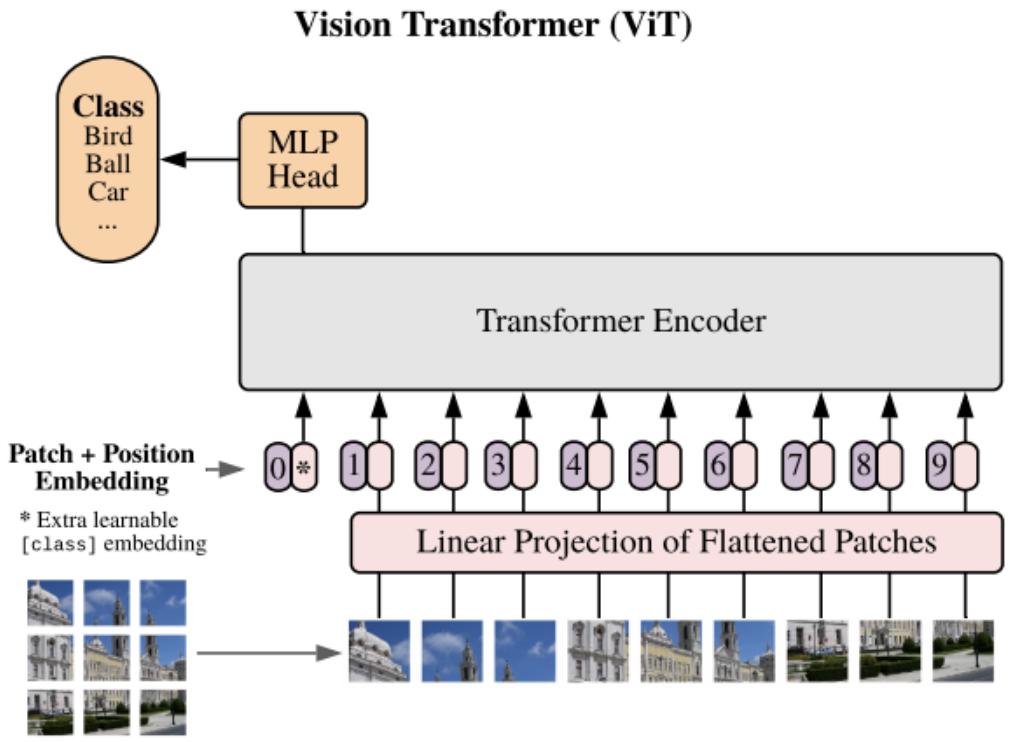

Figure 7. A Vision Transformer Model [9]

3.3. Challenges in Synchronization, Optimization, and Adaptation

Numerous applications [10], such as driverless cars, smart cities, healthcare monitoring, industrial automation, and robotics, become possible when CV and CPS are integrated. However, this integration has several significant challenges, especially when it comes to synchronizing and optimizing machine learning algorithms. These challenges are the following:

3.3.1. Synchronization Challenges

CPS requires synchronization since it involves several subsystems operating in real-time, frequently in dispersed contexts. The following difficulties arise while integrating a CV with CPS:

- Real-time Data Fusion: CV systems process visual data alongside other sensors like LiDAR, RADAR, and accelerometers. Poor decision-making may result from system lags or timestamp misalignments.
- Latency in Decision Making: The processing of deep learning-based CV algorithms is time-consuming, making real-time synchronization with CPS controls essential. Delays can compromise safety in systems like autonomous vehicles and drones.
- Distributed Processing: Coordinating CV tasks among nodes in a distributed CPS network is challenging, particularly while handling time-sensitive communications and preserving system dependability.

3.3.2. Optimization Challenges

Efficient CV algorithms are crucial for real-world CPS applications, but optimizing them poses significant hurdles, including:

- Resource Constraints: Memory and processing power on CPS devices, particularly edge devices, are frequently constrained. Because deep learning models require many resources, optimizing them within these limitations might be challenging.

 doi:10.20944/preprints202501.0521.v2

- Model Efficiency: Techniques like model compression and pruning are necessary to reduce the size and complexity of neural networks for tasks such as object detection and recognition on resource-constrained edge devices.
- Real-time optimization: There is a trade-off between time performance and accuracy, particularly challenging for low-latency applications like autonomous navigation.
- Communication Bandwidth: In distributed CPS, efficiently transmitting high-dimensional CV data requires methods such as video compression and local processing using edge computing.

3.3.3. Adaptation Challenges

CPS adopts flexible and adaptable CV algorithms under dynamic environments. Key challenges are the following:

- Dynamic Environments: CV algorithms must adapt continuously to changing conditions, such as variations in lighting, weather, and the presence of new obstacles, unlike static conditions.
- Transfer Learning and Domain Adaptation: It is challenging to adapt pre-trained models to new environments with minimal retraining, such as when autonomous vehicles move from urban to rural areas.
- Online Learning and Incremental Updates: CPS requires real-time model updates without requiring full retraining, which is computationally costly, due to continuous data streaming.
- Handling Uncertainty and Noise: To ensure accurate decision-making for managing noisy, incomplete, or uncertain sensor data, the method should be robust.

The relevant techniques and methods identified in the references to address these challenges are summarized in Table 1.

Table 1. Relevant techniques/methods to address the challenges

Challenges	Reference	Techniques/Methods
Synchronization		
Real-time Data Fusion	[11]	A hybrid framework integrating an FCNx and an EKF
	[12]	The notion of known templates method, using predefined patterns; An information-theoretic approach, leveraging statistical properties of the data.
	[13]	LFF YOLO Network
	[14]	The digital twin architecture integrates the different modules.
	[15]	Multi-sensor fusion algorithms
Latency in Decision Making	[13]	LFF YOLO Network
	[14]	The publish-subscribe pattern architecture
	[16]	Parallel Algorithm for Multitarget Tracking
	[17]	some form of coordinate transformation or homography to map 2D face coordinates onto the 3D space
Distributed Processing	[11]	A hybrid framework integrating an FCNx and an EKF
	[18]	A distributed motion control system for reconfigurable manufacturing systems
Optimisation		
Resource Constraints	[19]	Compressed MobileNet V3 Architecture
	[20]	Key optimisation techniques including distributed optimisation algorithms, gradient compression, and adaptive learning rate strategies
Model Efficiency	[21]	Pruning: Removes unimportant connections; Quantization: Reduces the number of bits representing each connection; Huffman Coding: weight sharing
	[15]	Dynamic power management; Efficient algorithms; Hardware optimisation
Real-time optimisation	[22]	A hyperheuristic multi-objective evolutionary search method and the best network hyperparameters
	[23]	CNN for mobile computer vision systems
	[20]	Federated learning and neuromorphic computing
Communication Bandwidth	[20]	Communication-efficient algorithms, including Ring All Reduce and decentralized training methods
	[15]	Open communication protocols
Adaptation		
Dynamic Environments	[24]	A physical-virtual interactive parallel light fields collection method
Transfer Learning and Domain Adaptation	[25]	Neural style transfer and GAN
	[26]	ResADM: a transfer-learning-based attack detection method
Online Learning and Incremental Updates	[27]	Model-Agnostic Meta-Learning and Conditional Neural Processes
Handling Uncertainty and Noise	[28]	Ensemble model

 doi:10.20944/preprints202501.0521.v2

4. Results and Analysis

4.1. Key Findings

4.1.1. CNN

CNN plays a dominant role in CV applications within CPS due to its specialized design for image analysis. Its layered architecture is highly effective at automatically learning patterns, features, and spatial hierarchies from images. This capability makes CNN exceptionally well-suited for image classification and object detection tasks.

CNN consists of several essential components: convolutional layers, which extract local features and spatial hierarchies; pooling layers, which perform downsampling to reduce dimensionality; and fully connected layers, which aggregate global features and enable decision-making. To integrate these elements, CNN uses flattening to convert the outputs of convolutional and pooling layers into a one-dimensional vector, serving as input for the fully connected layers. The architecture prioritizes parameter sharing, enabling efficient processing of visual data.

Owing to its innovative design, CNN has become instrumental in advancing image processing. They are especially powerful in visual understanding because of their ability to extract and process spatial features. Its impact extends beyond image processing to object detection, image classification, and semantic segmentation tasks. In CPS which integrates computational algorithms with physical processes, CNN provides the robust perception capabilities necessary for effective environmental interaction, solidifying its role as a cornerstone of modern CV. Table 2 illustrates recent CNN techniques for CV applications.

 doi:10.20944/preprints202501.0521.v2

Table 2. A Table for CNN Techniques of Computer Vision Applications

Reference	CNN Techniques / Models	Key Contributions	Main Tasks / Application Domains	Major Limitations
[29]	U-Net architecture, Diffusion models, Optical flow estimation, Image-to-image models, and Frame interpolation	It manages imperfections in flow estimation effectively and decoupled edit propagate design	Local edits and short-video creation	Dependence on the first frame and struggle with highly complex or rapid motions
[30]	ResNet, Dense Networks (DenseNet), Generative Adversarial Networks (GANs) and Multi-scale Networks	Advancing techniques	Video object detection for medical imaging, surveillance, and autonomous driving	Artificial degradation may not apply to real-world situations.
[11]	Fully convolution neural network (FCNx) for classification tasks and ResNet for feature extraction	Hybrid multi-sensor fusion uses encoder-decoder FCNx with extended Kalman Filter for environmental perception.	Environmental perception for autonomous driving	Significant computational resources
[31]	CNNs various model compression techniques	Comprehensive review	Mobile devices, edge computing, IoT and embedded systems	A trade-off between computation and performance
[23]	CNNs on TensorFlow and TensorRT platforms via parallelism	Comprehensive Latency Analysis, Novel Measurement Techniques, optimization Strategies, and Latency-Throughput Trade-Offs	Reducing latency in cloud gaming, optimizing AR and VR delay applications and strategies to object detection and recognition models	Sensor Dependency and significant computational resources
[32]	Sparse Polynomial Regression and Energy-Precision Ratio (EPR)	Predictive Framework: NeuralPower	Mobile Devices, Data Centres, and embedded systems	Specific GPU platforms and may not generalize well to all hardware configurations
[33]	Mask R-CNN: Extends Faster R-CNN, Region of Interest (RoI) Align, FCNs for the mask prediction	Instance Segmentation, accuracy improvements in pose estimations	Object Detection and Segmentation, human pose estimations, AR applications	Significant computational resources, performance depending on specific applications and datasets
[34]	Single Shot MultiBox Detector (SSD), uses of default boxes and multiscale feature maps in detecting objects	Unified framework: SSD for real-time detection	real-time object detection for autonomous driving, embedded systems, and AR applications	Significant computational resources, not well performance on very small objects
[35]	DEtection TRansformer (DETR): Combines a common CNN backbone with a transformer architecture.	End-to-End Object Detection and Bipartite Matching Loss	Object Detection in various applications: autonomous driving, surveillance, and robotics.; and Panoptic Segmentation	Significant computational resources, not well performance on very small objects
[36]	Pre-trained CNN model for image extraction and Truncated Gradient Confidence-Weighted (TGCW) Model for online classification	Improved accuracy and efficiency by noise handling	Image classification in medical imaging and personal credit evaluation	Significant computational resources and noise sensitivity
[14]	Preprocessing step using OpenCV, YOLOv5 for real-time object detection	Integrated Framework for precise position estimation and error levels below 1 degree and 3D rendering of vehicles and their surroundings in digital twin visualization	Accurate position estimations	inaccuracies in varying lighting or occlusion scenarios
[37]	3D Coordinate Mapping and Hybrid Reality Integration	Development of a Hybrid Reality-Based Driving Testing Environment	Autonomous Driving Development and extends to Internet of Vehicles	Reducing stability in higher frequency and incomplete real-world testing
[24]	Parallel light field platform, a data-driven approach for self-occlusion and inconsistency in viewpoints, colmap for offline re-construction	Improvements in PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index Measure) metrics.	Applications requiring accurate 3D modeling and relighting, such as virtual reality, game development, and visual effects	Variations in color temperature affecting 3D reconstruction and low-quality reconstruction models
[38]	Adaptive LFV coding and future integration with decentralized deep learning	balancing computation and communication latency to optimize performance	Enabling realistic digital twins, VR, AR and IoT-driven applications	Processing in off-line, not well performing in dynamic lighting conditions and occlusions
[39]	Integration of Yolov7 for human pose estimation and the DeepFace pre-trained model for age, gender, and race estimation	While Yolov7 performed well, the DeepFace model fell short in accuracy	The task of estimating human height from a single full-body image	Inaccurate performance in the DeepFace model and only single image input
[40]	EmoFusioNet, a deep fusion-based model	EmoFusioNet uses stacked and late fusion methods to ensure a color-neutral ER system, achieving high accuracy	A real-time facial emotion-based security	Underperformance for very dark-skinned individuals due to poor resolution of CMOS cameras

Recent advancements in object detection and image classifications have focused heavily on different approaches, Region-Based Convolutional Neural Network (R-CNN), Residual Network (ResNet) and You Only Look Once (YOLO). These methods are often benchmarked against datasets like Microsoft COCO and ImageNet [41].

R-CNN is a two-stage object detection model. It generates around 2,000 region proposals per image, resizes each, and processes them through separate networks for feature extraction and classification [41]. To improve efficiency, regions with significant overlap are discarded, keeping only the highest-scoring classified regions. However, this approach is computationally intensive. To address this, Fast R-CNN and Faster R-CNN were developed to streamline the process, reducing processing time and improving accuracy.

 doi:10.20944/preprints202501.0521.v2

Mask R-CNN, an extension of Faster R-CNN, adds a branch for instance segmentation, enabling the prediction of both bounding boxes and segmentation masks. This versatility allows it to handle tasks beyond object detection, such as human pose estimation while maintaining a relatively low computational overhead. Mask R-CNN operates at about 5 frames per second (fps) and is adaptable for other applications with minimal effort [33].

ResNet is a CNN architecture designed for feature extraction and image classification, with a primary focus on training deep neural networks efficiently without performance degradation, such as vanishing gradients. It employs residual learning with skip connections, enabling gradients to flow directly through the network. This innovation makes very deep networks, such as ResNet-50 and ResNet-101, both trainable and efficient. ResNet is widely used for tasks like image classification, image segmentation, and object detection, often serving as a backbone in detection models.

YOLO is a single-stage detector optimised for speed, making it ideal for real-time object detection. Unlike R-CNN, YOLO processes the entire image in a single pass through one network, generating fewer than 100 bounding box predictions per image [41]. Although faster, YOLO tends to have a higher localization error than R-CNN but produces fewer background false positives.

Several enhanced versions of the YOLO architecture, including YOLOv2, YOLOv3, YOLOv4, and YOLOv5, have been introduced to improve accuracy while retaining the high speed required for real-time applications. Though generally less accurate than Faster R-CNN, these versions are fast enough to meet the demands of real-time systems such as self-driving cars [42].

Other models, such as the Single Shot Multibox Detector (SSD), have been proposed as alternatives to YOLO, offering improvements in the network's backbone structure [34]. Simultaneously, innovations like focal loss have been introduced to replace traditional loss functions, enhancing detection accuracy.

4.1.2. Federated Learning

Federated Learning (FL) holds significant promise for synchronizing distributed CPS nodes because it trains models across multiple devices while keeping data localized. This approach enhances privacy and minimizes the need for centralized data storage, a critical advantage for sensitive applications.

CPS typically operates through a network of distributed devices, such as sensors, actuators, and edge devices, spread across various physical locations. FL enables these devices to collaboratively train a shared model without centralising data. By aggregating model updates instead of raw data, FL supports decentralized architectures, aligning models across nodes while maintaining data privacy.

Given that CPS involves distributed components requiring seamless coordination, FL provides a privacy-preserving and decentralized mechanism to synchronize these components effectively. This ensures synchronized decision-making and consistent behavior across the entire CPS network. FL also facilitates continuous learning, allowing devices to locally update models and periodically synchronize them. Such capabilities are crucial for real-time applications like autonomous vehicles and industrial robotics.

FL offers several advantages for CV applications in CPS [43]. The following are some key advantages.

- Privacy Preservation: FL retains data on local devices, sharing only model updates. This safeguards sensitive visual data, such as surveillance footage or medical records, addressing significant privacy concerns.
- Scalability: FL efficiently handles large-scale distributed systems, making it ideal for extensive CPS networks with numerous devices.
- Reduced Latency: Local data processing and updates minimize communication overhead and latency compared to centralized training methods.
- Heterogeneity Handling: FL can leverage adaptive aggregation techniques and personalized models to address the heterogeneity among nodes, ensuring synchronization is maintained even in diverse and resource-imbalanced environments.

 doi:10.20944/preprints202501.0521.v2

- Robustness and Adaptability: FL supports continuous learning and adapts to new data, enhancing the robustness of models in dynamic environments.

FL has two synchronization techniques[44]. They are the following.

- Synchronous FL: All nodes synchronize their updates simultaneously, which can be challenging due to varying computational capabilities and network conditions.
- Asynchronous FL: Nodes update the model independently, offering more flexibility and efficiency but potentially leading to stale updates.

FL faces several challenges that researchers are actively working to address. Here are some of the key challenges.

- Non-IID Data: Data from different nodes may not be identically distributed, which can affect model performance. Techniques like data augmentation and domain adaptation can help mitigate this issue.[45]
- Communication Overhead: Efficient communication protocols and compression techniques are essential to reduce the bandwidth required for model updates.[43]
- Model Heterogeneity: Different devices may have varying computational capabilities. Federated learning frameworks need to account for this by using adaptive algorithms that can handle heterogeneous environments.[43]

FL plays a pivotal role in CPS synchronization by facilitating decentralized collaboration, real-time adaptation, preservation of privacy, and scalability. It enables distributed devices to collaboratively train and synchronize models, effectively addressing CPS-specific challenges. This ensures efficient, reliable, and privacy-conscious coordination in modern smart systems.

4.1.3. Meta-Learning

Meta-learning in CV focuses on training models that can quickly adapt to new visual tasks with minimal data, computational effort, and dynamic scenarios. This is particularly useful in CV applications where tasks vary widely and data is scarce. Meta-learning techniques enable CV models to excel at tasks with very little labelled data, such as identifying new object classes from just a few examples. Meta-learned models can extract broadly applicable features, enabling rapid adaptation across diverse visual domains.

Meta-learning offers several techniques in the field of CV. Here are some key techniques.

- Prototypical Networks: These networks address the problem of few-shot classification by enabling generalization to new classes with only a few examples per class. They learn a metric space where classification is based on distances to class prototype representations. They offer a simpler inductive bias compared to other few-shot learning methods, producing excellent results with limited data. [46]
- Siamese Networks: These networks consist of twin neural networks that share parameters and weights. They are trained to maximize the distance between dissimilar pairs and minimize the distance between similar pairs. which consists of twin networks with shared weights trained to map similar observations close together in feature space and dissimilar ones farther apart. Experiments on cross-domain datasets demonstrate the network's ability to handle forgery across various languages and handwriting styles. [47]
- Model-Agnostic Meta-Learning (MAML): MAML algorithm is compatible with any model trained by gradient descent, applicable to tasks such as classification, regression, and reinforcement learning. The objective is to train a model on diverse tasks to generalize to new tasks with minimal training samples. This method optimises model parameters to enable rapid adaptation with just a few gradient steps on new tasks, making the model easy to fine-tune. MAML achieves state-of-the-art performance on few-shot image classification benchmarks, delivers strong results in few-shot regression, and accelerates fine-tuning in policy gradient reinforcement learning. [48]

 doi:10.20944/preprints202501.0521.v2

- Memory-augmented models: These models, such as Neural Turing Machines (NTMs), can enhance the efficient incorporation of new information without relearning their parameters by quickly encoding and retrieving new information. They can quickly assimilate data and predict accurately with only a few samples. Santoro et al., 2016 [49] introduce a novel method for accessing external memory that focuses on memory content, eliminating the dependence on location-based mechanisms used in previous approaches.

Meta-learning offers several advantages in the field of CV. The following are some key benefits.

- Fast Adaptation: Meta-learning enables models to quickly adapt to new tasks with minimal data. It is critical for dynamic applications, such as autonomous vehicles or drones operating in changing environments.
- Data Efficiency: By leveraging prior knowledge from related tasks, meta-learning reduces the need for extensive training data. This efficiency is crucial in applications like medical imaging, where annotated data is often scarce.
- Cross-Domain Learning: Meta-learning helps models generalize better across different tasks and domains. That facilitates adaptation across domains, such as transferring knowledge from medical imaging to aerial imagery. Google Vizier includes features such as transfer learning, which allow models to use knowledge from previously optimised tasks to accelerate and enhance the optimisation of new ones[50].
- Personalization: Meta-learning adapts models to individual preferences or environments, such as tailoring AR applications for unique users.

Meta-learning has numerous applications in CV to improve model performance and adaptability across various tasks. Here are some prominent examples.

- Image Classification: Meta-learning algorithms can quickly adapt to classify new categories of images with minimal data, and quickly recognize unseen classes in few-shot or zero-shot settings.
- Object Detection and Tracking: By leveraging prior knowledge, meta-learning models can enhance object detection and tracking capabilities, making them more robust to variations in the visual environment.
- Image Segmentation: Meta-learning can improve the performance of image segmentation tasks, where the goal is to partition an image into meaningful segments. This is particularly useful in medical imaging and autonomous driving.
- Facial Recognition: Meta-learning techniques can be used to develop facial recognition systems that adapt quickly to new faces with limited training data, enhancing security and personalization applications.
- Pose Estimation: Meta-learning can be applied to pose estimation tasks, where the model needs to predict the pose of objects or humans in images. This is useful in the fields of robotics and augmented reality.
- Scene Understanding: Meta-learning allows CV systems to interpret new or unseen scenes for applications such as navigation or augmented reality (AR).

Meta-learning in CV faces several challenges that researchers are actively striving to overcome. Here are some notable challenges.

- Scalability: Meta-learning algorithms often struggle with scalability when applied to large-scale datasets and high-dimensional data typical for CV tasks. Efficiently scaling these algorithms while maintaining performance is a significant challenge.
- Generalization: Ensuring that meta-learning models generalize well across a wide range of tasks and domains is difficult. Models trained on specific tasks may not perform well on unseen tasks, highlighting the need for better generalization techniques.
- Computational Complexity: Meta-learning methods can be computationally intensive, requiring significant resources for training and adaptation. This complexity can limit their practical application, especially in resource-constrained environments.

 doi:10.20944/preprints202501.0521.v2

- Data Efficiency: When meta-learning aims to be data-efficient, achieving this in practice can be challenging. Models often require a careful balance between leveraging prior knowledge and adapting to new data with minimal samples.
- Task Diversity: The diversity of tasks used during meta-training is crucial for the model's ability to generalize. However, creating a sufficiently diverse set of tasks that accurately represent real-world scenarios is challenging.
- optimization Stability: Ensuring stable and efficient optimization during the meta-training phase is another challenge. Meta-learning models can be sensitive to hyperparameters and the choice of optimization algorithms.
- Interpretability: Meta-learning models, especially those based on deep learning, can be difficult to interpret. Understanding how these models make decisions and adapt to new tasks is important for trust and transparency.

4.2. Themes and Categories

4.2.1. Synchronization Strategies

Synchronization refers to aligning the timing and interaction between various subsystems, sensors, and actuators within a CPS. In the context of ML-based computer vision, a list of synchronization strategies is the following:

- Timestamping: Timestamping involves attaching precise time metadata to each data packet as it is generated, enabling the alignment and correlation of data streams from heterogeneous sources. Yang and Kupferschmidt [51] implement timestamp synchronization specifically for video and audio signals, demonstrating its effectiveness. This approach is typically simpler and less computationally intensive compared to more complex synchronization methods.
- Sensor Fusion: This technique is widely used in embedded systems to integrate data from multiple sensors, providing a more accurate and reliable representation of the environment. It is commonly applied in areas such as autonomous vehicles, robotics, and wearable devices. [11] introduce a real-time hybrid multi-sensor fusion framework that combines data from cameras, LiDAR, and radar to enhance environment perception tasks, including road segmentation, obstacle detection, and tracking. The framework employs a Fully Convolutional Neural Network (FCN) for road detection and an Extended Kalman Filter (EKF) for state estimation. Designed to be cost-effective, lightweight, modular, and robust, the approach achieves real-time efficiency while delivering superior performance in road segmentation, obstacle detection, and tracking. Evaluated on 3,000 scenes and real vehicles, it outperforms existing benchmark models.
  Moreover, Robyns et al.[14] demonstrate how to communicate from the physical system to the digital twin for visualizing the industrial operation by using Unreal Engine. The digital twin features a modular architecture based on the publish-subscribe pattern, enabling the integration of multiple data processing modules from heterogeneous data streams.
- Real-time task scheduling: This technique involves orchestrating machine learning and computer vision tasks to ensure timely and reliable operations. CPS applications, such as autonomous vehicles, robotics, and smart manufacturing, demand low-latency, high-accuracy processing while operating under strict deadlines and resource constraints as illustrated in Figure 8.

 doi:10.20944/preprints202501.0521.v2

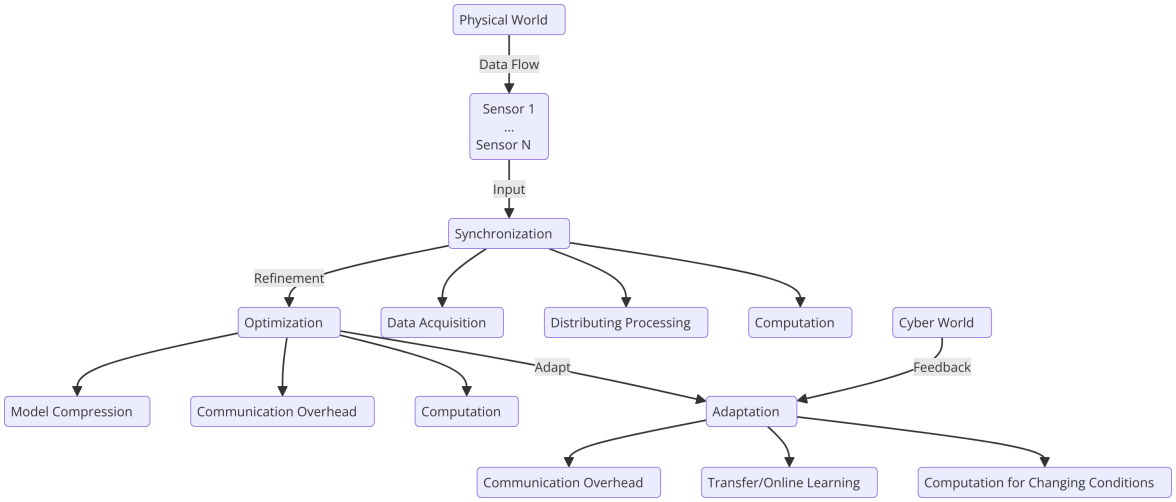

Figure 8. Synchronization, Optimization and Adaptation in computer vision in CPS.

Hu et al.[52] propose a framework to enhance the efficiency of AI-based perception systems in applications like autonomous drones and vehicles. The framework focuses on prioritizing the processing of critical image regions, such as foreground objects, while de-emphasizing less significant background areas. This strategy optimizes the use of limited computational resources. The study leverages real LiDAR measurements for rapid image segmentation, enabling the identification of critical regions without requiring a perfect sensor. By resizing images, the framework balances accuracy and execution time, offering a flexible approach to handling less important input areas. This method avoids the extremes of full-resolution processing or completely discarding data. Experiments are conducted on an AI-embedded platform with real-world driving data to validate the framework's practicality and efficiency.

4.2.2. Optimization Approaches

Balancing computational efficiency and accuracy is a critical challenge when applying ML techniques to CV within CPS. CPS systems are often constrained by limited computational resources (such as low-power embedded devices), real-time processing requirements, and the need for high accuracy in tasks like object detection, tracking, segmentation, and decision-making. Below are several optimization approaches that can help strike a balance between these competing demands:

- Model Compression Techniques: Techniques[31] such as pruning, quantization, knowledge distillation, low-rank factorization, and transfer learning are applied to reduce the size of deep learning models without sacrificing significant performance. This is particularly critical for edge devices and CPS with limited hardware resources [53] and [19].
  - Pruning: Reducing the number of neurons or connections in a neural network by removing weights that have little influence on the output. This decreases the size of the model, making it computationally more efficient without significantly sacrificing accuracy.
  - Quantization: Reducing the precision of the weights and activations in the model, from 32-bit floating-point to 8-bit integer or even binary. This leads to reduced memory footprint and faster computation, especially on specialized hardware (like FPGAs and GPUs).
  - Deep compression: Han, Mao, and Dally [21] introduce "deep compression", a three-stage pipeline (pruning, quantization, and Huffman coding) designed to reduce the storage and computational demands of neural networks, enabling deployment on resource-constrained embedded systems. Pruning removes unnecessary connections, reducing the number of connections by 9× to 13×. Quantization enforces weight sharing, reducing the representation of each connection from 32 bits to as few as 5 bits. Huffman coding further compresses the quantized weights. Experiments on AlexNet showed a 35× reduction in weight storage, with VGG-16 and LeNet achieving 49× and 39× reductions, respectively, while maintaining accu-

 doi:10.20944/preprints202501.0521.v2

racy. This compression enables these networks to fit into on-chip SRAM cache, significantly reducing energy consumption compared to off-chip DRAM access. The approach enhances the feasibility of deploying complex neural networks in mobile applications by addressing storage, energy efficiency, and download bandwidth constraints.

- Knowledge Distillation: A process where a smaller, less complex "student" model learns to approximate the outputs of a larger, more complex "teacher" model. This can yield a more computationally efficient model with a similar accuracy. Hinton, Vinyals, and Dean [54] demonstrate the effectiveness of distillation successfully transferring knowledge from ensembles or highly regularized large models into a smaller model. On MNIST, this method works well even when the distilled model's training set lacks examples of certain classes. For deep acoustic models, such as those used in Android voice search, nearly all performance gains from ensembles can be distilled into a single, similarly sized neural net, making deployment more practical. For very large neural networks, performance can be further improved by training specialist models that handle highly confusable class clusters. However, distilling the knowledge from these specialists back into a single large model remains an open challenge. This approach highlights the potential of distillation to balance performance and efficiency in machine learning systems.
- Low-rank factorization - This reduces the number of parameters in deep learning models by approximating weight matrices with lower-rank matrices. This technique helps in compressing models and speeding up training and inference. Cai et al.[55] propose a joint function optimisation framework to integrate low-rank matrix factorization and a linear compression function into a unified optimisation approach, designed to reduce the number of parameters in DNNs, computational and storage costs while preserving or enhancing model accuracy.
- Transfer learning is a machine learning method that involves reusing a model trained on one task to solve a related task. This approach allows the model to leverage its prior knowledge, enabling it to learn new tasks effectively even with limited data. In CPS applications, transfer learning minimizes the need for extensive manual labeling by transferring insights from similar domains. By utilising models pre-trained on large-scale datasets (e.g., ImageNet) as a foundation, transfer learning avoids the need for training from scratch. Fine-tuning only a few layers enables CPS systems to adapt quickly to new tasks or environments, significantly reducing computational costs.
  Bird et al.[22] explore unsupervised transfer learning between Electroencephalography (EEG) and Electromyography (EMG) using both MLP and CNN approaches. The models were trained with fixed hyperparameters and a limited set of network topologies determined through a multi-objective evolutionary search. Identical mathematical features were extracted to ensure compatibility between the networks. Their research demonstrates the application of cross-domain transfer learning in human-machine interaction systems, significantly reducing computational costs compared to training models from scratch.

- Lightweight Architectures: Use specialized architectures designed for efficiency while maintaining good accuracy. These include models like MobileNet and EfficientNet, which are designed to run efficiently on resource-constrained devices.
  - MobileNet is a class of efficient models designed for mobile and embedded vision applications. Howard et al.[56] utilize a streamlined architecture with depthwise separable convolutions to create lightweight deep neural networks. Two global hyperparameters are introduced to balance latency and accuracy, enabling model customization based on application constraints. Extensive experiments show that MobileNets perform well compared to other popular models on ImageNet classification. Their effectiveness is demonstrated across diverse applications, including object detection, fine-grain classification, face attribute analysis, and large-scale geo-localization.

 doi:10.20944/preprints202501.0521.v2

  - EfficientNets are a family of CNNs designed to achieve high accuracy with significantly improved computational efficiency. They were introduced as a solution to the challenge of scaling CNNs while balancing resource usage and performance. Tan and Li[57] propose a compound scaling method, a simple and effective approach for systematically scaling up a baseline CNN while maintaining efficiency under resource constraints. Using this method, the EfficientNet models achieve state-of-the-art accuracy with significantly fewer parameters and FLOPS, and high performance on both ImageNet and five transfer learning datasets, demonstrating their scalability and efficiency.

- Hardware Acceleration and optimisation: The method often involves leveraging parallelism (e.g., through graphics processing units (GPUs) or specialized hardware like tensor processing units (TPUs)) or optimising the inference pipeline to speed up processing as illustrated in Figure 9.

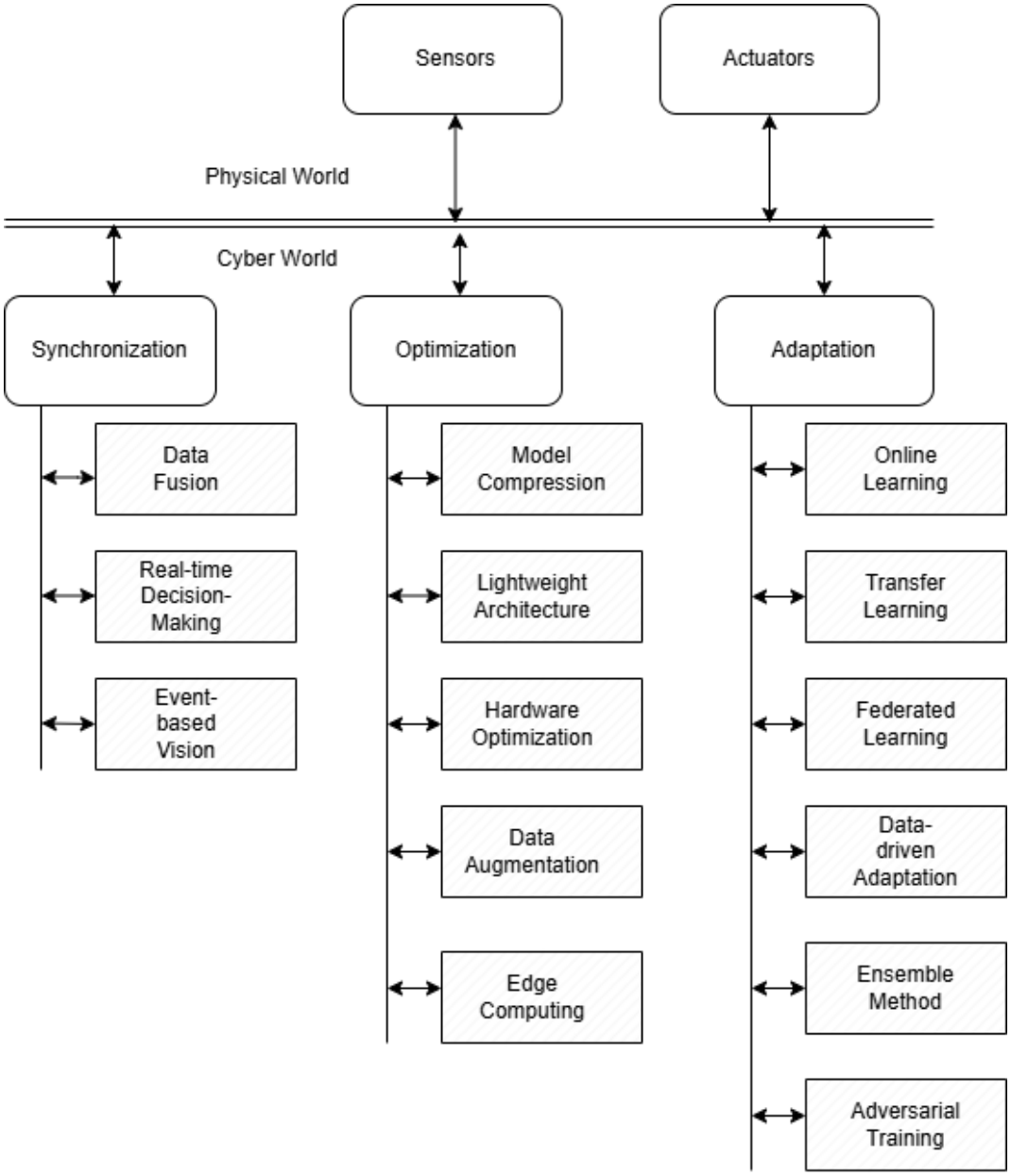

Figure 9. A Overview of the Proposed Framework.

Since 2015, distributed-memory architectures with GPU acceleration have become the standard for machine learning workloads due to their growing computational demands[53]. Maier et al.[16] depict a GPU implementation of the parallel auction algorithm, optimised for both open

 doi:10.20944/preprints202501.0521.v2

computing language (OpenCL) and compute unified device architecture (CUDA) environments, which reduces memory usage and increases speed compared to previous implementations, making it ideal for embedded systems with large problem sizes. Experimental results across two GPUs and six datasets show a best-case speedup of 1.7x, with an average speedup of 1.24x across platforms. Additionally, this approach meets strict real-time requirements, especially for large-scale problems, as demonstrated in sensor-based sorting applications. However, optimisation is further constrained by fixed initial parameters, such as GPU architecture or model accuracy, limiting flexibility for future adjustments. Different GPUs deliver varied performance depending on factors, like batch size and execution context. Achieving optimal performance requires careful balancing trade-offs between accuracy, throughput, and latency[23].

- Data Augmentation: Data augmentation involves applying various transformations (such as rotation, scaling, and cropping) to the training dataset, thereby artificially expanding its size and diversity. This approach helps enhance the performance of smaller models. In many real-world scenarios, collecting sufficient training data can be challenging. Data augmentation [58] addresses this issue by increasing the volume, quality, and variety of the training data. Techniques for augmentation include deep learning-based strategies, feature-level modifications, and meta-learning approaches, as well as data synthesis methods using 3D graphics modeling, neural rendering, and generative adversarial networks (GANs).
  - Deeply Learned Augmentation Strategies: These techniques use deep learning models to generate augmentations automatically, improving the diversity and quality of the data. Neural networks are employed to create realistic data variations, thus enhancing the model's robustness.
  - Feature-Level Augmentation: This method modifies specific features of the data, rather than the raw image itself. Common operations include changing attributes like contrast, brightness, or texture. Such adjustments can improve the model's ability to generalize across different scenarios.
  - Meta-Learning-Based Augmentation: Meta-learning approaches focus on learning how to generate useful augmentations based on the characteristics of the data. These methods aim to optimise the augmentation strategy itself, improving the model's learning efficiency across various tasks.
  - Data Synthesis Methods: These involve generating synthetic data through techniques like 3D graphics modeling. This approach creates realistic data variations, which is particularly useful for simulating rare or hard-to-capture events in real-world scenarios.
  - Neural Rendering: This technique uses neural networks to generate images from 3D models or abstract representations, producing realistic augmentations that can improve the diversity and realism of the training data.
  - Generative Adversarial Networks (GANs): GANs are employed to create synthetic data by training two competing networks—the generator and the discriminator. The generator produces new images, while the discriminator evaluates their authenticity. GANs can generate highly realistic augmentations, significantly boosting the dataset's diversity.
- Edge Computing: This paradigm involves moving computational tasks closer to the data source, such as on embedded devices at the network's edge. By processing data locally, edge computing reduces the latency associated with transmitting data to and from remote servers, enabling real-time responses critical for applications like autonomous navigation and real-time surveillance. This approach also conserves bandwidth and enhances data privacy. Significant improvements in latency and throughput have been observed when deploying trained networks on mobile devices and remote servers [23].
  Deng et al.[59] expand the scope of edge computing by integrating it with AI into a concept called Edge Intelligence, categorized into AI for Edge and AI on Edge:

 doi:10.20944/preprints202501.0521.v2

- AI for Edge: Utilizes AI technologies to address key challenges in edge computing, such as optimising resource allocation, reducing latency, and managing data efficiently.
- AI on Edge: Focuses on performing the entire AI lifecycle, including model training and inference, directly on edge devices.

In distributed learning, the model is trained collaboratively across multiple edge devices, with only model updates—rather than raw data—being transmitted to a central server. This approach reduces communication bandwidth requirements and enhances data privacy. Tron and Vidal [60] demonstrate the application of distributed computer vision algorithms, highlighting that the storage requirements at each node depend solely on local data and remain constant irrespective of the number of cameras involved. For accelerating deep learning training, integrating distributed architectures with techniques such as gradient compression and adaptive learning rates is essential [20].

4.2.3. Adaptation Mechanisms

CPSs often operate in dynamic and unpredictable environments. Machine learning models must be adaptable to new conditions or evolving system requirements. Here are key adaptation mechanisms to ensure robust performance:

- Data-driven Adaptation: This approach involves leveraging data to enable models or systems to adjust and optimise their performance in response to dynamic conditions or specific challenges. In Shen et al.'s studies[24], the parallel light field platform supports the collection of realistic datasets that capture diverse lighting conditions, material properties, and geometric details. These datasets empower data-driven adaptation by providing models with inputs that closely mimic real-world scenarios, ensuring robust generalization across varying environments. To handle self-occlusion, the conditional visibility module adopts a data-driven strategy, dynamically computing visibility along rays based on input viewpoints. Instead of relying on predefined rules, the module learns and predicts visibility directly from data, enabling it to adapt effectively to diverse viewing conditions. Moreover, data-driven techniques are applied to address specular reflection challenges and depth inconsistencies, showcasing the system's capability to adapt to complexities arising from changing viewpoints. These adaptations, powered by data, enhance the model's ability to adjust predictions under varying environmental and geometric conditions. Another example is presented in Kaur et al.'s article[25], where data augmentation techniques are used to generate variations in the dataset, allowing models to learn from a wide range of scenarios. This helps models adapt to unseen conditions during inference. The techniques discussed include Geometric Transformations, Photometric Transformations, Random Occlusion, and Deep Learning-based Approaches. The choice of augmentation methods depends on the nature of the dataset, the problem domain, and the number of training samples available for each class.
- Online Learning: This approach involves continuously updating a model with new labeled or pseudo-labeled data collected during deployment. In machine learning, models must learn and adapt in real time as fresh data becomes available. This is especially crucial in CPS where the system must adjust to changes such as varying lighting conditions for cameras or evolving cyber-security threats. Implementing online learning in production environments typically requires several steps: debugging offline, continuous model evaluation, managing data drift, performing regular offline retraining, using efficient algorithms, ensuring data quality, having a rollback plan, and applying incremental updates[27].
  For online learning, Hu et al.[36] introduce the pre-trained Truncated Gradient Confidence-weighted (Pt-TGCW) model, which combines offline and online learning techniques for tasks like image classification. This model highlights the effectiveness of incremental learning approaches. Additionally, Lu et al.[61] propose Passive-Aggressive Active (PAA) learning algorithms, which update models using misclassified instances and leverage correctly classified examples with low

 doi:10.20944/preprints202501.0521.v2

confidence. Their methods enhance performance across various online learning tasks, including binary and multi-class classification.

- Transfer Learning: This approach involves leveraging pre-trained models on large datasets and fine-tuning them for specific tasks, utilising existing knowledge to improve robustness. In CPS, models trained on one dataset may need to be adapted to different environments or contexts. TL enables this adaptation by fine-tuning pre-trained models with smaller, task-specific datasets, making it easier to adjust models to new situations. This is particularly important in CPS, where models must be trained in one context and then applied to another. For instance, Wang et al.[26] propose a transfer-learning approach for detecting attacks in CPS using a Residual Network (ResNet). Their method refines source model parameters through an intentional sampling technique, constructing distinct sample sets for each class and extracting relevant features from attack behaviors. This approach results in a robust network capable of accurately detecting attacks across different CPS environments.
- Ensemble Methods: The method combines multiple models to enhance prediction accuracy and reliability, addressing the weaknesses of individual models. The ensemble model proposed by Tahir et al.[28] incorporates diverse architectures (MobileNetV2, Vgg16, InceptionV3, and ResNet50), each capable of adapting to different features or patterns within the dataset. These models may excel in recognizing distinct aspects of the data, and their combination allows the system to handle a wider range of scenarios and data variations, such as differences in X-ray image quality or fracture types. By aggregating predictions from multiple models, the ensemble approach adapts to changes in data quality and characteristics, improving robustness and generalization. This is particularly important when working with medical datasets like Mura-v1.1, where data can vary in terms of noise, resolution, and imaging conditions. Preprocessing techniques such as histogram equalization and feature extraction using Global Average Pooling further support adaptation, helping the model adjust to variations in image quality. These methods ensure that the model can effectively handle different input characteristics. The combination of diverse architectures and preprocessing techniques in the ensemble model enhances its adaptability, robustness, and accuracy, which is crucial for reliable performance in the complex and variable field of medical image analysis.
- Adversarial Training: This technique enhances the model's robustness by making it more resistant to small, intentional perturbations in the input data that could otherwise lead to misclassifications. By generating adversarial examples [62] and incorporating them into the training process, the model learns to recognize and correctly classify inputs that would typically confuse it, thus improving its generalization capability. This approach provides insights into how neural networks can adapt to better resist adversarial perturbations, ultimately strengthening their robustness. By using adversarial examples during training, the model becomes more adaptable to a wider range of input variations, making it more resilient and capable of generalising effectively across different datasets, architectures, and training conditions.

  Another example[63] involves handling adversarial perturbations through randomized smoothing, which strengthens a model's robustness against adversarial attacks by adding Gaussian noise to the input data. This technique ensures the model is "certifiably robust" to adversarial perturbations, enabling it to maintain reliable performance even when confronted with modified inputs. Training the model with both original and noise-augmented data enhances its capacity to generalize across varied conditions, including adversarial scenarios. This adaptation process equips the model to handle a broader range of input variations, increasing its resilience to unforeseen changes in data distribution. As a formal adaptation technique, randomized smoothing ensures stability and high performance, even under adversarial conditions. By incorporating noise during training, this method significantly bolsters the model's ability to manage adversarial inputs, enhancing its robustness and generalization in challenging environments.

 doi:10.20944/preprints202501.0521.v2

- Federated Learning: In distributed CPS, where devices are spread across different locations (e.g., smart cities, industrial IoT), FL allows individual devices to train models locally and share updates, improving model performance across the system without centralising sensitive data.
  In Himeur's article[43], FL is used to distribute computational tasks across multiple clients, alleviating the load on central servers and enabling collaborative machine learning while ensuring data privacy. FL employs various aggregation methods, such as averaging, Progressive Fourier, and FedGKT while incorporating privacy-preserving technologies like Secure Multi-Party Computation (MPC), differential privacy, and homomorphic encryption to safeguard sensitive information. Despite its advantages, FL in Computer Vision (CV) encounters several challenges, including high communication overhead, diverse device capabilities, and issues related to non-IID (non-independent and identically distributed) data, complicating model training and performance consistency.
  To lower resource constraints, Jiang et al.[64] introduce a Federated Local Differential Privacy scheme, named Fed-MPS (Federated Model Parameter Selection). Fed-MPS employs a parameter selection algorithm based on update direction consistency to address the limited resource issue in CPS environments. This method selectively extracts parameters that improve model accuracy during training while simultaneously reducing communication overhead.

5. Discussions

5.1. Critical Evaluation

5.1.1. Increased Focus on Real-Time Performance

Time synchronization protocols are essential for aligning clocks across devices in a CPS, ensuring consistent timestamps for images and sensor data. This consistency is critical for achieving temporal coherence in ML-based CV tasks [29,30,51]. Recent advances in research highlight the importance of real-time synchronization [11,36], which enhances the reliability and efficiency of CPS applications, including autonomous vehicles, industrial automation, and robotics.

Innovative synchronization algorithms [36] have demonstrated improvements in data consistency across devices and networks in CPS. These advancements aim to reduce noise in classification sample data, increase the accuracy of modern classifiers, and achieve faster convergence speeds. Real-time data fusion, which integrates information from multiple synchronized sensors (e.g. cameras, LiDAR, and radar), further enhances CV tasks such as object detection and tracking [11,16,30]. Additionally, edge computing is increasingly used for local synchronisation, enabling efficient and timely processing.

A notable example is the two-level synchronisation mechanism presented in a real-time distributed 3D human pose estimation (HPE) platform for human-machine interaction systems in industrial environments [65]. This approach addresses communication challenges like delay and bandwidth variability, demonstrating superior accuracy and scalability compared to state-of-the-art methods and marker-based infrared motion capture systems. Real-time optimisation techniques are designed to reduce computational demands while addressing the scalability and complexity of neural networks. Model compression methods [21,54,55] have been effective in simplifying neural architectures, while lightweight architectures [56,57] provide efficient solutions for real-time inference. Hardware acceleration, including parallel and distributed computing, has also been pivotal in managing large-scale data processing for real-time image analysis in extensive CPS networks. GPU-based implementations [16,23] optimise performance by minimizing latency and increasing throughput. Edge computing further minimizes latency by reducing the need to transmit data to centralized servers, enhancing both processing efficiency and data privacy [59,66]. Additional techniques, such as real-time data augmentation [25,58], involve dynamic transformations during preprocessing to prepare data effectively at runtime. Computational load in video processing is also mitigated through strategies like frame skipping or adaptive sampling.

 doi:10.20944/preprints202501.0521.v2

5.1.2. Hybrid Methods

Hybrid methods that integrate physical and virtual optimisation layers are emerging as powerful approaches to improve system efficiency and robustness in CPS. These methods leverage the strengths of both the physical domain (e.g., real-world sensors, actuators, and processes) and the virtual domain (e.g., simulations, predictive algorithms, and digital twins) to create a cohesive and adaptive system.

Digital Twins are a key method in advancing operations across various domains. A bio-inspired LIDA (Learning Intelligent Distribution Agent) cognitive-based Digital Twin architecture [67] facilitates unmanned maintenance of machine tools by enabling self-construction, self-evaluation, and self-optimisation. This architecture provides valuable insights into implementing real-time monitoring in dynamic production environments. In the manufacturing industry, Digital Twins enhance flexibility and efficiency while addressing safety and reliability challenges in collaborative tasks between human operators and heavy machinery. They enable accurate detection and action classification under diverse conditions, as demonstrated in studies [14,17]. Another prominent application involves an Autonomous Driving test system under hybrid reality [37], which improves efficiency, reduces costs, and enhances safety, offering a robust solution for autonomous driving development.

Digital twins within the Metaverse can replicate physical CPS environments, facilitating real-time monitoring and decision-making. Rehman et al. [66] explore a system that identifies patients' emotions using image processing techniques within a virtual environment, where avatars represent both patients and physicians. As virtual reality (VR) and augmented reality (AR) technologies continue to evolve, they are expected to create increasingly immersive experiences. The study [66] encourages researchers and practitioners to explore the integration of technology with psychological therapy, aiming to validate this innovative approach and establish a foundation for future research.

However, the current maturity of Digital Twin technology often necessitates offline system halts for model updates, with implementations relying on backends that impose strict data exchange requirements. To address these challenges, the CoTwin framework [68] introduces a dynamic approach that allows online model refinement in CPS without disrupting operations. This framework leverages a blockchain-based collaborative space for secure data management and integrates neural network algorithms for fast, time-sensitive execution. It ensures stable, efficient performance while meeting the temporal requirements of CPS, offering a competitive edge in industrial applications.

5.1.3. Human-in-the-Loop

Human-in-the-loop is a prominent approach in CPS, particularly in areas where human decision-making, oversight, or intervention is essential. By integrating humans into the control loop, this approach enables real-time interaction, supervision, and system adjustments driven by human input. While challenging to implement, advancements in digital technologies have greatly facilitated this integration. Studies [69–71] emphasize the importance of human involvement in the control loop, showcasing its benefits in real-time system interaction and adaptability. This methodology is crucial for various manufacturing applications, such as assembly tasks, quality control, decision-making support, and health risk assessments, ensuring enhanced safety, flexibility, and operational efficiency. Moreover, the human-in-the-loop paradigm extends to other fields like decentralized traffic merging and highway lane merging systems [72], where it significantly improves system performance and safety outcomes.

5.1.4. Standardized Benchmarks

CPS applications require precise synchronization and robust optimisation to function effectively. However, developing standardized benchmarks for evaluating and comparing CPS solutions poses significant challenges. Below is a discussion of the key challenges and their consequences.

- Diversity in Application Requirements: CPS applications have highly varied requirements in terms of latency, fault tolerance, and real-time responsiveness. For example, autonomous driving systems require low latency and strict real-time synchronization [30], whereas construction

 doi:10.20944/preprints202501.0521.v2

operations prioritize robustness and fault tolerance [14]. These differences make it difficult to create universal benchmarks that address the needs of all domains effectively.

- Heterogeneous Architectures: CPS systems involve a complex mix of hardware, software, and communication protocols. Variability in processing speeds, sensor accuracies, and network latencies requires synchronization and optimization solutions customized to diverse architectures. Standard benchmarks often fail to account for these architectural disparities.
- Dynamic Operating Environments: CPS must perform reliably in environments with unpredictable changes, such as varying workloads, communication delays, and environmental disturbances. Creating benchmarks that accurately simulate such dynamic conditions is a complex and resource-intensive task that makes standardization challenging.

The following implications will be produced.

- Inconsistent Performance Metrics: Without common benchmarks, researchers and practitioners rely on ad hoc evaluation methods. This inconsistency makes it challenging to compare the efficiency, scalability, and effectiveness of different synchronization and optimization techniques.
- Limited Reproducibility: The absence of standardized frameworks impedes reproducibility, as the experimental setup and evaluation criteria vary widely between studies. This inconsistency hinders progress in developing reliable CPS solutions.
- Barriers to Collaboration: Standardized benchmarks foster collaboration by providing a shared foundation for evaluating CPS technologies. Without them, it becomes difficult for researchers, engineers, and domain experts to collaborate effectively within a cohesive ecosystem.
- Challenges in Real-World Applications: Many CPS applications, such as automotive systems and smart grids, require rigorous testing and validation to meet safety and performance standards. The lack of standardized benchmarks hampers this process, potentially affecting system reliability and trustworthiness.

Addressing the challenges outlined above would enable consistent performance evaluation, promote reproducibility, and encourage collaboration between disciplines. In addition, establishing robust benchmarks would improve the reliability and safety of CPS in real-world applications, contributing to the development of reliable and efficient systems.

5.2. Interdisciplinary Perspectives

Cyberspace technology has seamlessly integrated into our modern world, underscoring the transformative synergy between ML, CV, and CPS. This interplay emphasizes the critical role of interdisciplinary collaboration in addressing complex challenges and driving technological innovation. Collaborative efforts among computer scientists, engineers, and domain experts are essential to harness the full potential of these technologies. The key points of this collaboration include:

5.2.1. Complexity of Interdisciplinary Challenges

The integration of ML, CV, and CPS presents intricate, domain-specific challenges that demand expertise across multiple disciplines. ML algorithms must be customized to account for the real-time behaviors characteristic of CPS, while CV models need to be designed to accurately interpret and analyze the physical environment. Engineers and computer scientists play a critical role in implementing these models within the physical constraints of systems, whereas domain experts provide the necessary context for their application. Rehman et al.'s research [66] highlights this interdisciplinary collaboration, where virtual reality (VR) and augmented reality (AR) technologists work alongside psychologists to assess patients' conditions, exemplifying how each field contributes unique insights to address these complex challenges.

5.2.2. Designing Effective Solutions

ML and CV technologies must be tailored to meet the objectives and constraints of CPS applications. While computer scientists design algorithms for tasks like object detection, engineers are

 doi:10.20944/preprints202501.0521.v2

tasked with integrating these algorithms into physical systems capable of real-time responsiveness. Domain experts ensure the system adheres to specific industry standards. For instance, in Wu et al.'s study [13], computer scientists develop algorithms for weak defect detection, engineers deploy these algorithms into production lines that operate in real-time, and domain experts validate the system's compliance with industry requirements and standards. Collaborative efforts are essential to create effective solutions that address both technical and domain-specific challenges.

5.2.3. Data Interpretation and Real-World Implementation

The raw data collected from sensors in CPS, such as cameras or LIDAR, requires effective processing and interpretation using CV and ML techniques. Computer scientists and engineers focus on developing algorithms and system architectures, while domain experts ensure the data are interpreted within the context of the specific real-world application. They guarantee the system responds appropriately to achieve outcomes like safety, performance, or efficiency. For instance, in assembly operations [73], a digital architecture integrates multiple sensors to monitor and improve the well-being of assembly operators. ML algorithms analyze this data to automatically assess the Ergonomic Assembly Worksheet, emphasizing factors like posture, applied forces, and material handling.

5.2.4. Real-Time Decision-Making

In CPS, particularly in applications like autonomous driving or robotics, real-time decision-making is essential. ML algorithms created by computer scientists for perception and decision-making must work seamlessly with virtual leader systems to analyze sensor data. In the article by Yedilkhan et al [74], ML models improve obstacle avoidance strategies through learned behaviours from prior data to handle uncertainty and adapt to dynamic environments. Engineers ensure that these systems are optimised for real-time performance and reliability. Collaboration with domain experts ensures that the systems are not only accurate but also safe, efficient, and compliant with industry standards.

The development of ML and CV systems for CPS is an ongoing process that requires constant feedback. Domain experts can provide valuable insights from real-world testing, helping engineers and computer scientists fine-tune algorithms. Collaboration enables continuous improvement by ensuring that the system is iteratively refined to address new challenges and incorporate emerging technologies.

5.3. Emerging Trends

5.3.1. Edge Artificial Intelligence

Edge Artificial Intelligence (AI) is a groundbreaking computing paradigm designed to perform machine learning model training and inference directly at the network edge [75]. This paradigm enables two distinct approaches [59]: AI on edge, where models are trained and inferred either collaboratively through direct interaction between edge devices or using local edge servers near these devices, and AI for edge, which focuses on integrating artificial intelligence into edge computing architectures. This integration enhances edge devices' ability to handle complex data processing and decision-making tasks. Although relatively new, the field has experienced remarkable growth recently, driving innovative CPS applications.

- Real-Time Processing and Low Latency: Edge AI revolutionizes real-time decision-making processes by enabling on-device data processing, which minimizes latency and ensures instant responses. This capability is indispensable for applications that demand immediate and reliable decision-making, such as autonomous vehicles and health care. In these scenarios, rapid responses are not only beneficial but also critical. For example, automotive vehicle systems require handling vast amounts of heterogeneous data from various sensors, requiring high-performance and energy-efficient hardware systems to process this information in real-time, interacting between functional modules seamlessly with low overhead, and facing strict energy constraints, emphasizing the need for optimised hardware and computational techniques. By decentralising intelligence, edge AI brings ML model training and inference directly to the network, enabling

 doi:10.20944/preprints202501.0521.v2

communication between edge systems and infrastructure, and reducing the computational burden on the edge systems [15].

Edge AI is a transformative technology that brings numerous benefits to the functionality and efficiency of medical devices, especially in the realm of the Internet of Medical Things (IoMT) [76]. By processing data locally, Edge AI ensures faster, real-time decision-making, crucial in medical contexts. For instance, in remote monitoring systems, critical health alerts can be instantly generated and communicated to caregivers or medical professionals, improving the reliability and responsiveness of these systems. In such cases, local storage capacities and synchronisation of sensor data may cause challenges to the application creators.

- Enhanced Security and Privacy: Edge AI minimizes the need to transmit sensitive data to central servers, significantly enhancing the security and privacy of decentralized CPS applications. This localized processing not only reduces exposure to potential data breaches but also strengthens the overall resilience of the system. Ensuring the reliability, security, privacy, and ethical integrity of edge AI applications is paramount, as edge devices handle sensitive information with potentially severe consequences in the event of a breach. Robust encryption methods, stringent access controls, and secure processing and storage frameworks are indispensable for safeguarding data and maintaining trust [75]. Hardware-supported Trusted Execution Environments are often employed to enhance security by isolating sensitive computations. However, these solutions present challenges related to performance and integration, necessitating a delicate balance between maintaining robust security and ensuring efficient system operations. Addressing these challenges is critical for the successful deployment of edge AI in secure and decentralized CPS environments.
- Energy Efficiency: The growing demand for AI applications highlights the need for energy-efficient and sustainable edge AI algorithms. Advanced AI, particularly deep learning, consumes substantial energy, posing sustainability challenges. Developing lightweight and energy-efficient AI models is essential for supporting edge devices with limited computational resources, thereby enhancing the sustainability of CPS applications. Computational offloading is another effective method to reduce energy consumption in edge devices [76].

  However, achieving a balance between high performance and energy efficiency is crucial. Often, small gains in accuracy require significantly more energy, which is inefficient and environmentally unsustainable when ultrahigh accuracy is not necessary. Researchers must carefully evaluate the trade-offs between accuracy and energy use.

  For the significant impact of energy consumption during the operation, production, and lifecycle of edge devices, creating durable, upgradeable, and recyclable devices is vital to minimize ecological impact. Implementing policies to promote energy-efficient AI and regulating the environmental footprint of device manufacturing and disposal are critical steps toward achieving sustainability in edge AI [75].
- Interoperability: Efforts are being made to develop comprehensive standards and frameworks to ensure seamless interoperability between edge devices and CPS components across diverse applications. These standards aim to establish uniform protocols for data exchange, device communication, and system integration, enabling heterogeneous edge devices and CPS components to work together cohesively. This interoperability is critical for supporting scalability, reducing system fragmentation, and fostering a more unified ecosystem that can accommodate advancements in hardware and software technologies.

  Moreover, the development of such frameworks addresses challenges related to compatibility, security, and system resilience, providing a robust foundation for reliable decentralized operations. These initiatives also incorporate mechanisms to manage dynamic environments, where edge devices and CPS components must adapt to changing conditions in real-time while maintaining performance and reliability.

 doi:10.20944/preprints202501.0521.v2

5.3.2. Self-Adaptive Systems Leveraging Reinforcement Learning (RL)

Self-adaptive systems are pivotal in addressing the dynamic and uncertain demands of modern technology landscapes. These systems adjust their behaviour autonomously to maintain optimal performance despite changes in their environment or internal state. While traditional approaches to adaptation rely on predefined rules or models created during design time, these methods struggle to cope with the unpredictable and complex nature of real-world environments. Reinforcement learning (RL) has emerged as a transformative solution, empowering self-adaptive systems with the ability to learn, adapt, and optimise decisions dynamically.

- Addressing Design-Time Uncertainty: One of the most significant challenges in developing self-adaptive systems is the uncertainty inherent at design time. Online RL provides a compelling solution [77]. By enabling systems to learn directly from interaction with their environment, RL equips self-adaptive systems with the ability to respond effectively to previously unencountered conditions. This adaptive capacity is critical for systems deployed in dynamic environments, such as autonomous vehicles or distributed cloud-edge networks, where operational contexts can shift unpredictably.
- Real-Time Decision-Making: The ability to make real-time decisions is the cornerstone of self-adaptive systems. RL excels in this domain by continually refining its policies based on operational feedback, ensuring the system remains responsive to changes. RL-driven systems autonomously optimize their behaviour, balancing competing objectives such as performance, energy efficiency, and reliability [77]. This capability is particularly valuable in applications like IoT-driven healthcare, where immediate responses to patient data can be life-saving, and in autonomous systems, where split-second decisions are vital for safety.
- Enhancing Efficiency: Efficiency is a critical consideration in the operation of self-adaptive systems. RL supports this by enabling dynamic resource allocation and optimizing the use of computational, energy, and network resources based on current demands. Deep RL integrates energy optimization with load balancing strategies, in order to minimize energy consumption while ensuring server load balance under stringent latency constraints [78]. Furthermore, RL's ability to handle nonlinear and stochastic environments makes it particularly well-suited for real-world applications, where unpredictability and instability are the norm. This adaptability ensures robust performance in dynamic and challenging conditions, reinforcing its utility across various domains.
- Generalization and Scalability: Deep RL extends the capabilities of RL by integrating neural networks to represent the learned knowledge. This allows self-adaptive systems to generalize their learning to unseen states and handle high-dimensional input spaces, such as sensor data or video streams. This generalization capability is crucial for scalability, enabling RL-driven self-adaptive systems to operate effectively in diverse and complex environments. Applications such as smart cities, where systems must manage vast amounts of real-time data from interconnected devices, benefit immensely from Deep RL's scalability and adaptability.

5.3.3. Hybrid Machine Learning Models

The rapid advancements in machine learning have led to the emergence of hybrid models that combine DL with traditional algorithms to achieve improved efficiency, flexibility, and scalability in diverse applications. These hybrid approaches aim to harness the strengths of both paradigms while mitigating their respective limitations.

- Enhanced Performance: Deep learning excels at extracting high-level features from unstructured data, such as images and text. However, it often requires significant computational resources. Traditional algorithms handle structured data and provide clear interpretability [79]. In [80], authors applied CNN and autoencoders to extract features and then followed by the particle swarm optimisation (PSO) algorithm to select optimal features and reduce dataset dimensionality while maintaining performance. Finally, the selected features were classified by the third stage

 doi:10.20944/preprints202501.0521.v2

using learnable classifiers decision tree, SVM, KNN, ensemble, Naive Bayes, and discriminant classifiers to process the acquired features to assess the model's correctness. Combining these techniques results in models that deliver high performance without the prohibitive costs of standalone deep learning methods.

- Improved Generalization: Hybrid models combine the strengths of deep learning and traditional algorithms, capitalising on deep learning's ability to handle complex, non-linear relationships in data while utilising traditional methods to enhance interpretability and generalization, particularly in scenarios involving smaller datasets. For example, the Adaptive Neuro-Fuzzy Inference System (ANFIS), as discussed in [80], exemplifies a hybrid network where fuzzy logic intuitively models nonlinear systems based on expert knowledge or data. Neural networks complement this by introducing adaptive learning capabilities, enabling the system to optimise parameters such as membership functions through input-output data. This integration empowers ANFIS to effectively model complex, nonlinear relationships, making it highly applicable in tasks such as prediction, control, and pattern recognition.
- Scalability and Adaptability to Diverse Tasks: Hybrid models offer remarkable flexibility, enabling customization for specific applications by integrating the most advantageous features of distinct paradigms. In [81], by combining Statistical Machine Translation (SMT), which uses statistical models to derive translation patterns from bilingual corpora, with Neural Machine Translation (NMT), which employs Sequence-to-Sequence (Seq2Seq) models with RNNs and dynamic attention mechanisms, these approaches capitalize on the statistical precision of SMT and the contextual richness of neural networks. Additionally, ensemble methods enhance translation quality further by amalgamating multiple models, proving particularly effective for domain-specific adaptations and ensuring robust performance.
- Limitations: Hybrid learning systems offer robust solutions for complex data-driven challenges by combining the strengths of both methodologies. However, they face several challenges [79], including high model complexity, which complicates configuration, optimization, and interpretation. Despite advances in transparency, their layered architecture often obscures decision-making processes, raising issues of interpretability. The extensive and diverse datasets required for training pose significant privacy and security risks. Additionally, deploying and maintaining these systems is resource-intensive due to their sophisticated architecture and the need for regular updates to stay aligned with evolving data and technologies. Real-time processing capabilities can be hindered by the computational intensity of DL components, and the energy demands of training and operating hybrid models raise environmental concerns. Long-term maintenance further demands substantial effort to ensure these models remain effective and relevant in dynamic environments.
- Future Research: Future research in hybrid learning should focus on deeper interdisciplinary integration with fields like cognitive science, medical, and computing to achieve AI systems that more closely emulate human cognition. Advancing model generalization is equally critical, emphasizing the development of adaptive systems capable of autonomously adjusting to varying datasets and environmental conditions. Additionally, enhancing AI accessibility is essential to democratize its use, improved educational resources, and community-driven initiatives, thereby broadening the impact of AI as a universal problem-solving tool [79].

5.4. Research Gaps

The integration of ML techniques, particularly in CV, within CPS has unlocked significant advancements in fields such as autonomous vehicles, smart cities, and industrial automation. These systems rely heavily on synchronization methods to ensure that distributed components collaborate effectively. However, the scalability of these synchronization methods remains a critical challenge, compounded by insufficient attention to real-world deployment issues and the lack of adaptive models for handling diverse and dynamic CPS environments. This subsection explores the key research gaps that hinder progress in this domain and highlights directions for future work. Identifying these

 doi:10.20944/preprints202501.0521.v2

research gaps is critical to improving the development of an efficient and resilient CPS. Some vital areas are outlined below.

5.4.1. Limited Scalability of Synchronization Methods

- Resource constraints in CPS: One of the most prominent issues with current synchronization methods is their limited scalability in CPS environments. These systems often operate under resource-constrained conditions, with devices such as sensors, cameras, and actuators constrained by bandwidth, energy, and computational power. Many existing synchronization techniques assume abundant resources, which is unrealistic in practical CPS deployments. We expect to have methods to balance the accuracy of synchronization with the energy efficiency and computational cost. Consequently, there is a need for lightweight synchronization algorithms that optimize resource usage without compromising accuracy or efficiency.
  In [82], the results show an improvement in precision with an increasing sampling rate at the cost of increased memory consumption and computation time. Similarly, in [12], post-deployment processing to align and synchronize data streams introduces computational overhead, which can challenge resource-constrained systems with limited processing power or memory. Moreover, in [18], the synchronization approach is based on standard components, which may have limitations in terms of precision and robustness, especially when scaling up or requiring higher performance.
- Bottlenecks in distributed systems: Distributed systems, another core aspect of CPS, face significant bottlenecks in synchronization due to the communication overhead and latency associated with global updates. This is particularly problematic in real-time applications like autonomous vehicles, where even minor delays can have critical consequences. One possible solution shown in [83] is the use of a polychronous model of computation for concurrent systems to free programming from synchronous timing models and to enhance robustness against clock synchronization failure-based attacks. This approach allows processes to execute and communicate at their paces without requiring rigid synchronization, thereby reducing bottlenecks caused by contention for shared resources.
  However, current research has not sufficiently addressed techniques to minimize communication requirements, such as using model pruning, gradient sparsification, or local aggregation. Becker et al. [84] show that contention among system modules severely affects latency and performance predictability, but LiDAR-related components contribute significantly to system latency and even high-end CPU and GPU platforms cannot achieve real-time performance for the complete end-to-end system. Furthermore, ensuring a balance between local computation and global model updates remains an unresolved challenge. Hybrid synchronization techniques that adapt dynamically to the system's real-time state could address this issue, but their development is still in its infancy.

5.4.2. Insufficient Focus on Real-World Deployment Challenges

- Environmental Variability: The real-world deployment of ML-based synchronization methods in CPS introduces a host of challenges that have not received sufficient attention. One major issue is the variability of real-world environments, which often include unpredictable network latency, device failures, and dynamic workloads. Current synchronization methods are not robust enough to handle these variations, and research on fault-tolerant approaches that can recover gracefully from such disruptions is limited. Developing methods that maintain performance despite environmental variability is essential to advance the reliability of CPS.
- Deployment at scale: Many synchronization methods are tested in controlled environments or small-scale settings, which do not reflect the challenges of real-world CPS deployments involving hundreds or thousands of nodes. In [67], the datasets and scenarios used are relatively simple and may not fully demonstrate the generality of the proposed cognitive Digital Twin architecture. More experiments in real industrial maintenance scenarios are needed to validate performance in multi-task, resource-allocation contexts involving personnel, spare parts, and

 doi:10.20944/preprints202501.0521.v2

materials. Additionally, the example focuses on updating microservices and the knowledge graph post-data analytics, rather than the physical-world operational responses. Future iterations could incorporate deviations between expected and actual outcomes into the self-evolution process to develop more realistic maintenance solutions.

- Real-time constraints: Real-time constraints further complicate deployment. Many CPS applications, such as surveillance and industrial automation, require synchronization methods that can operate in real-time to process high-frequency data streams. However, the latency introduced by current synchronization methods makes them unsuitable for such applications. Research on event-driven or asynchronous synchronization mechanisms that prioritize low-latency processing is still nascent and demands further exploration.

5.4.3. Adaptive Models for Diverse CPS Environments

- Heterogeneity in devices: The diversity of CPS environments presents another significant research gap. These systems often involve a wide range of devices with varying capabilities, such as sensors, drones, and cameras, each with different levels of processing power, storage, and communication bandwidth. Current synchronization methods are not designed to account for this heterogeneity, leading to inefficiencies in resource utilization. Adaptive synchronization algorithms that dynamically adjust to the capabilities and constraints of individual nodes are needed to address this gap.
- Diverse tasks: CPS tasks vary widely, ranging from object detection to anomaly detection and action recognition. Each task has unique synchronization requirements, but current methods often adopt a one-size-fits-all approach, failing to optimize for the priorities of individual tasks. Developing task-specific synchronization strategies and exploring multi-tasking synchronization approaches could significantly enhance the performance and flexibility of CPS.
- Dynamic environments: Dynamic environments pose further challenges, as CPS systems often operate under non-stationary conditions where data distributions, network topologies, or operational requirements can change over time. Existing models lack the adaptability to handle such conditions effectively. Self-learning synchronization methods that adjust based on feedback from the environment offer a promising direction for future research, enabling models to remain robust and effective in evolving CPS scenarios.

5.4.4. Integration with Emerging Technologies

- Edge and Federated Learning: Edge and federated learning paradigms have shifted the focus from centralized to decentralized systems, necessitating new synchronization strategies tailored for these frameworks. Efficient edge-to-cloud synchronization techniques and privacy-preserving methods for federated learning are critical areas that require further investigation.
- Neuromorphic Computing and Event-Based Vision: These techniques are transforming data processing in CPS by introducing asynchronous, event-driven paradigms. These technologies demand synchronization methods that can handle irregular and spiking data streams, but current research has not kept pace with these advancements. Developing synchronization techniques compatible with neuromorphic hardware and spiking neural networks could unlock new possibilities for CPS applications.

6. Recommendations

6.1. For Practitioners

This study explores strategies and techniques to synchronize, optimise, and adapt ML models within CPS. Despite their potential, deploying ML models in CPS presents unique challenges, including the demands for real-time processing, security, and reliability across varied physical environments. Successful deployment requires leveraging advanced methodologies such as edge computing to enable low-latency applications and federated learning to ensure secure, distributed data processing. When

 doi:10.20944/preprints202501.0521.v2

combined with other practical implementation tasks, these approaches can substantially improve the robustness and efficiency of CV systems in CPS.

6.1.1. Implementing Edge Computing

Edge computing plays a critical role in enabling real-time decision-making for CPS applications such as autonomous vehicles, industrial automation, and surveillance systems. By processing data locally on edge devices instead of relying on centralized cloud servers, edge computing significantly reduces latency and enhances responsiveness. This approach is vital for high-stakes applications where low latency is essential, and it also minimizes bandwidth usage and reduces reliance on stable internet connectivity. The following key tasks are fundamental to a successful implementation of edge computing in CPS:

- Model optimisation: Apply techniques [19,31,53] such as pruning, quantization, and knowledge distillation to develop lightweight models that operate effectively in resource-constrained edge devices without sacrificing accuracy.
- Hardware Utilization: Select hardware platforms designed specifically for edge computing, such as NVIDIA Jetson, Intel Movidius, or Google Coral. Additionally, leverage accelerators such as GPUs and TPUs to improve computational efficiency.
- Runtime Frameworks: optimised inference frameworks, including TensorRT, ONNX Runtime, or PyTorch Mobile, are used to ensure efficient and reliable execution of models on edge devices.
- Partitioning Workloads: Distribute tasks strategically between edge devices and the cloud. Execute time-sensitive computations on the edge while offloading resource-intensive processes, such as retraining or extensive analytics, to cloud infrastructure.
- Real-Time Monitoring: Develop mechanisms to continuously monitor the performance and health of edge devices, ensuring reliability and consistent operation even in varying environmental conditions.

6.1.2. Leveraging Federated Learning

FL is a decentralized method of training machine learning models that enables edge devices to collaborate without requiring the sharing of raw data. This approach is particularly advantageous in CPS where data privacy and security are critical, such as in healthcare, smart cities, and defense applications. FL not only enhances data security by keeping sensitive information local but also facilitates the creation of models that are representative of diverse environments, thereby improving generalization and reducing bias. Key tasks for implementing FL effectively include the following:

- Data Locality: Maintain sensitive data on local devices and transfer only model updates to a central server. This approach minimizes the risk of data breaches and ensures compliance with privacy regulations such as GDPR.
- Communication Efficiency: Employ techniques like model compression, update sparsification, and asynchronous communication to reduce the bandwidth required for transmitting model updates between devices and the central server.
- Security Measures: Implement safeguards such as differential privacy and secure multi-party computation to protect data and model parameters from adversarial attacks and ensure the integrity of the learning process.
- Federated Averaging: Utilize aggregation algorithms, like FedAvg, to effectively combine model updates from multiple devices and enhance the robustness of the aggregation process by integrating outlier detection or Byzantine-resilient methods to handle potentially malicious updates.
- Heterogeneity Handling: Design systems capable of accommodating a wide range of edge devices with varying computational capabilities and network conditions. This can be achieved by dynamically allocating tasks based on each device's capabilities, ensuring efficient and equitable participation in the learning process.

 doi:10.20944/preprints202501.0521.v2

6.1.3. Integration of Edge Computing and Federated Learning with CPS

Seamless integration of edge computing and federated learning with CPS is essential to ensure harmonious operation between machine learning (ML) models and the system's physical components. This integration supports real-time decision-making, sensor fusion, fault tolerance, and scalability. We discuss key factors influencing effective integration in the below:

- Monitoring and Continuous Learning: To remain effective in dynamic environments, edge computing, and federated learning systems require ongoing monitoring and adaptability. Key elements include:
  - Performance Metrics Tracking: Continuously monitor model performance metrics, such as latency, accuracy, and confidence, to promptly detect and address potential issues.
  - Periodic Updates and Retraining: Incorporate mechanisms for regular model updates and retraining with newly collected data to maintain accuracy and relevance.
  - Anomaly Detection: Implement systems to flag anomalous data or behaviour, enabling swift intervention when deviations from expected operations occur.
- Security and Privacy: Robust security and privacy measures are vital for safeguarding data and ensuring trust in CPS operations. These measures include:
  - End-to-End Encryption: Secure all data transmissions between edge devices, cloud servers, and central aggregation points using encryption.
  - Role-Based Access: Restrict access to models, data, and system components based on user roles and authentication protocols.
  - Defenses Against Adversarial Attacks: Employ strategies like input sanitization, adversarial training, and anomaly detection to protect against malicious activities.
- Ethical and Regulatory Compliance: Ethical standards and regulatory requirements are critical for public trust and the lawful deployment of ML models in CPS. Key considerations include:
  - Transparency: Provide clear and transparent documentation and explanations of model operations, especially in safety-critical applications.
  - Standards Compliance: Ensure adherence to relevant standards and laws, for example, ISO 26262 for automotive safety [85] and GDPR for data protection [86], to guarantee ethical and legal deployment.

6.2. For Researchers

CPS frequently operates on edge devices with limited computational resources, memory, and energy capacity. This constraint is especially critical in domains like IoT devices, autonomous drones, and wearable technologies, where real-time decision-making and prolonged operation are essential. To address these challenges, lightweight ML models must be designed to effectively balance computational efficiency and accuracy.

6.2.1. Developing Lightweight ML Models for Resource-Constrained CPS

Approaches such as model compression, pruning, quantization, and knowledge distillation (discussed in Section IV.B.2) are promising for reducing model size and complexity [87]. These techniques help optimise resource utilization without significantly compromising model performance. Future research can delve deeper into these methodologies, emphasizing the need to maintain interpretability and robustness in constrained environments.

A particularly promising tool for designing resource-efficient ML models is Neural Architecture Search (NAS). NAS automates the development of high-performing neural networks tailored to specific resource constraints, requiring minimal human intervention. Its framework comprises three core components [88,89]:

 doi:10.20944/preprints202501.0521.v2

- Search Space (SSp): Defines the range of architectures that NAS can explore. Advances in SSp have broadened the scope of candidate designs, enabling the discovery of innovative architectures that were previously unattainable.
- Search Strategy (SSt): Encompasses methods for exploring the defined search space. Recent research has focused on improving the efficiency of search strategies, optimising the balance between computational resources and performance outcomes.
- Validation Strategy (VSt): Refers to the techniques used to evaluate the performance of candidate architectures. Enhanced validation strategies have increased the reliability of NAS results while minimizing the time and resources required for evaluation.

Although NAS is still in its early stages, it holds immense potential. Its applications are expected to extend beyond image classification into domains requiring complex network designs, such as multi-objective optimization, model compression, and advanced tasks like object detection and semantic segmentation [85].

Moreover, the integration of NAS with emerging technologies like federated learning and edge computing presents exciting possibilities. These synergies could enable real-time, distributed model optimisation, fostering the development of scalable and adaptive ML systems. By leveraging such advancements, NAS can facilitate the creation of robust, efficient, and resource-aware ML models that are well-suited for dynamic CPS environments. [89].

In conclusion, developing lightweight ML models tailored for resource-constrained CPS is a critical research direction. By combining techniques like model optimization with automated solutions such as NAS, the field can achieve breakthroughs in efficiency, scalability, and adaptability, paving the way for innovative CPS applications.

6.2.2. Exploring TL for Faster Adaptation to New Tasks

Transfer learning is a machine learning approach where knowledge acquired in solving a source task is applied to a related but different target task. By utilizing pre-trained models or previously learned knowledge, transfer learning accelerates learning, reduces dependence on extensive labelled data in the target domain, and enhances performance, particularly in data-scarce or computation constraints.

Transfer learning provides a promising solution in CPS, which often operate in dynamic environments or face conditions different from their original training scenarios. By enabling models to adapt rapidly to new tasks or domains using pre-trained knowledge, TL minimizes the need for large, annotated datasets and computationally intensive retraining. However, domain shift, where the source and target domains have different data distributions, remains a significant challenge [90]. For example, a model trained on clean, curated datasets might perform poorly on noisy, real-world data. Overcoming domain shift requires methods like domain adaptation to reduce the distributional differences between the source and target domains. [91] and [92] have made significant progress but exhibit certain limitations (no learning features adaptively from the source and using a discontinuous decision strategy). Recent techniques introduced by [93] enhance outcomes by identifying and leveraging transferable structures in high-dimensional settings, offering robust theoretical guarantees and empirical benefits. Other innovative approaches [90,94] include :

- Few-Shot Learning: Facilitates robust training with minimal data in the target domain.
- Zero-Shot Learning: Uses cross-domain knowledge to predict unseen target classes without any labeled data.
- Generalization: Focuses on transferring knowledge across related tasks (e.g., object detection and semantic segmentation) to save retraining time and resources.
- Federated Transfer Learning (FTL): Extends TL to decentralized, privacy-sensitive contexts where source and target data cannot be shared or centralized [95].

 doi:10.20944/preprints202501.0521.v2

6.3. For Policy Suggestions

Developing individual automotive CPS using best-effort technologies is feasible but fraught with challenges due to technical complexity, high costs, and the potential for errors. To address these issues and support CPS development, several policies and guidelines can be implemented:

6.3.1. Establish Standardized Benchmarks for ML Performance in CPS

Standardized benchmarks are essential for evaluating and comparing machine learning models used in CPS. They ensure consistent, reliable performance metrics aligned with real-world applications. However, no such repository exists for automotive CPS. The recommendations are below.

- Create comprehensive benchmark repositories to include typical and worst-case models.
- Mitigate IP and security concerns by forming regulatory bodies to redact and standardize real-world models.
- Develop industry-specific benchmarks tailored to the needs of various CPS domains.
- Mandate performance evaluations against benchmarks to ensure baseline performance and reliability prior to deployment.

6.3.2. Promote Open-Source Tools

Open-source tools are transformative, fostering innovation, accessibility, and collaboration while lowering costs. The key benefits are:

- Collaboration: Enable global developers to share ideas and improvements, driving innovation and creative problem-solving.
- Accessibility: Make CPS development affordable by eliminating proprietary software costs, leveling the playing field for all.
- Transparency: Build trust through open review, auditing, and enhancement of code, promoting ethical and reliable systems.
- Accelerated Development**: Leverage pre-trained models and ready-to-use tools to reduce development time significantly.

6.3.3. Encourage Cross-Domain Collaboration

CPS development spans diverse sectors, benefiting from shared knowledge and multidisciplinary expertise. The recommendations are:

- Foster research programs that integrate engineering, computer science, economics, and social sciences.
- Establish CPS innovation hubs to tackle common challenges like security, real-time data processing, and model generalization.
- Create industry consortia to set shared goals, develop common standards, and address deployment challenges collaboratively.

6.3.4. Regulate and Promote Ethical Use of CPS

CPS interacts with physical processes and human lives, necessitating ethical deployment, privacy protection, and transparency. The recommendations are:

- Develop ethical guidelines and regulations to ensure CPS operates safely and fairly, especially in sensitive sectors, like healthcare and autonomous vehicles.
- Support explainable CPS for transparency in machine learning models, fostering public trust and accountability.
- Enforce robust cybersecurity and privacy standards to safeguard physical assets and data against attacks or misuse.
- Promote diversity in research and development teams to create inclusive CPS technologies that consider diverse societal impacts

 doi:10.20944/preprints202501.0521.v2

7. Conclusion

7.1. Summary of Findings

7.1.1. CNN

CNNs are essential in CV applications within CPS due to their ability to automatically learn spatial features and patterns from images. CNNs excel in tasks like image classification and object detection through their layered architecture, which includes convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully connected layers for decision-making. Parameter sharing and efficient visual data processing make CNNs foundational to modern CV applications. Recent advancements in CNN-based methods include:

- R-CNN: A two-stage object detector with high accuracy but computationally intensive. Variants like Fast R-CNN, Faster R-CNN, and Mask R-CNN improve efficiency and extend functionality to instance segmentation and pose estimation.
- ResNet: Focuses on training deep networks efficiently using residual learning and skip connections, widely applied in image classification, segmentation, and as a backbone for detection models.
- YOLO: A single-stage real-time detector that processes images in one pass, offering high speed with moderate localization errors. Versions like YOLOv2–YOLOv5 enhance accuracy while maintaining real-time performance, making them suitable for applications like autonomous vehicles.
- SSD: Alternatives to YOLO that improve detection accuracy and address specific limitations in network design and loss functions.

7.1.2. Federated Learning

FL is a transformative approach for synchronizing distributed CPS by enabling collaborative model training across multiple devices while keeping data localized. This decentralized methodology enhances privacy, reduces the need for centralized data storage, and supports continuous learning, making it ideal for sensitive, large-scale, and real-time applications like autonomous vehicles and industrial robotics. Key advantages of FL for CPS include:

- Privacy Preservation: Retains data on local devices, sharing only model updates to protect sensitive information.
- Scalability: Efficiently handles extensive distributed networks with numerous devices.
- Reduced Latency: Minimizes communication overhead by processing data locally.
- Heterogeneity Handling: Adapts to diverse and resource-imbalanced environments.
- Robustness and Adaptability: Supports continuous learning and dynamic updates to maintain model reliability.

FL employs two synchronization techniques:

- Synchronous FL: Updates are synchronized simultaneously, challenging in heterogeneous environments.
- Asynchronous FL: Updates occur independently, improving flexibility but risking stale updates.

Challenges include handling non-IID data across nodes, reducing communication overhead, and addressing computational disparities among devices. Despite these, FL's ability to facilitate decentralized collaboration, real-time adaptation, and privacy-conscious coordination makes it pivotal for CPS synchronization and scalability in modern smart systems.

7.1.3. Meta-Learning

Meta-learning in CV focuses on creating models capable of quickly adapting to new tasks with minimal data and computational resources. It is particularly beneficial for tasks with scarce data, enabling models to generalize across diverse domains and applications. Meta-learning techniques emphasize extracting broadly applicable features for rapid adaptation, making it suitable for dynamic

 doi:10.20944/preprints202501.0521.v2

scenarios like autonomous vehicles, medical imaging, and augmented reality. Key Meta-Learning Techniques are following:

- Prototypical Networks: Facilitate few-shot classification by learning a metric space where classification is based on distances to class prototypes.
- Siamese Networks: Twin networks that map similar data points closer in feature space, are useful for tasks like forgery detection across languages and styles.
- Model-Agnostic Meta-Learning: Trains models to adapt quickly to new tasks with a few gradient steps, excelling in few-shot classification, regression, and reinforcement learning.
- Memory-Augmented Models: Use external memory mechanisms to rapidly encode and retrieve new information without extensive retraining.

The advantages of Meta-learning in CV are:

- Fast Adaptation: Quickly adapts to new tasks with limited data, essential for dynamic environments like drones or robotics.
- Data Efficiency: Leverages prior knowledge, reducing the need for extensive labeled data, critical in areas like medical imaging.
- Cross-Domain Learning: Enhances generalization across different visual domains, aiding in knowledge transfer between tasks.
- Personalization: Tailors models to individual preferences or unique environments, such as user-specific AR applications.

The applications of Meta-learning in CV include:

- Image Classification: Quickly identifies unseen categories with few-shot or zero-shot learning.
- Object Detection and Tracking: Enhances robustness to visual variations.
- Image Segmentation: Useful in medical imaging and autonomous driving.
- Facial Recognition: Adapts to new faces with minimal training data.
- Pose Estimation and Scene Understanding: Critical for robotics and AR applications.

Challenges in Meta-Learning:

- Scalability: Difficulty in handling large-scale datasets and high-dimensional CV tasks.
- Generalization: Struggles to perform well on unseen tasks and domains.
- Computational Complexity: High resource requirements can limit applicability in constrained environments.
- Task Diversity: Developing diverse task sets for training is challenging but essential for real-world generalization.
- optimisation Stability: Sensitive to hyperparameters and optimisation methods, requiring careful tuning.
- Interpretability: Deep meta-learning models lack transparency, complicating trust and usability.

7.1.4. Synchronization in CPS

Synchronization is critical for aligning the timing and interactions among subsystems, sensors, and actuators in CPS. In ML-based CV, the following strategies are used:

- Timestamping: Attaches precise time metadata to data packets to align heterogeneous data streams.
- Sensor Fusion: Integrates data from multiple sensors for accurate environmental representation, used in applications like autonomous vehicles and robotics.
- Real-Time Task Scheduling: Ensures low-latency, high-accuracy processing under strict resource and time constraints, crucial for autonomous vehicles, drones, and robotics.

These strategies collectively enhance the synchronization and efficiency of CPS in ML-based CV applications.

 doi:10.20944/preprints202501.0521.v2

7.1.5. Optimisation Approaches

In CPS, balancing computational efficiency and accuracy is key due to constraints like limited hardware resources, real-time processing needs, and high-accuracy demands. Various optimisation approaches help manage these challenges:

- Model Compression Techniques:
  - Pruning: Removes redundant neurons or connections to reduce model size without significantly impacting accuracy.
  - Quantization: Lowers the precision of model weights, reducing memory usage and speeding up computation.
  - Knowledge Distillation: Transfers knowledge from a large, complex model (teacher) to a smaller, simpler one (student), maintaining similar accuracy while enhancing computational efficiency.
  - Low-rank Factorization: Approximates weight matrices with lower-rank matrices, reducing parameters, speeding up training, and improving inference efficiency.
  - Transfer Learning: Reuses pre-trained models for related tasks, reducing the need for extensive data labeling and speeding up adaptation to new tasks.
- Lightweight Architectures:
  - MobileNet: Uses depthwise separable convolutions to create lightweight models, customizable for different trade-offs between latency and accuracy.
  - EfficientNets: Achieve high accuracy with fewer parameters and FLOPS, designed for efficient scaling while maintaining performance.
- Hardware Acceleration and optimisation:
  - Parallelism: Utilizes GPUs and specialized hardware (e.g., TPUs) for faster processing, crucial for large-scale problems.
  - Inference Pipeline optimisation: Streamlines processing to meet real-time requirements, balancing trade-offs between accuracy, throughput, and latency.
- Data Augmentation:
  - Deeply Learned Augmentation: Uses deep learning models to generate data variations, enhancing robustness.
  - Feature-Level Augmentation: Alters specific data features (e.g., brightness, contrast) to improve generalization.
  - Meta-learning: Learns to generate optimal augmentations based on data characteristics.
  - Data Synthesis: Using techniques like 3D modeling and GANs to generate synthetic data, increasing the diversity of the training set.
- Edge Computing:
  - Edge Intelligence: Combines AI and edge computing to address challenges such as reducing latency, optimising resources, and enhancing data privacy.
  - Distributed Learning: Collaborative model training across edge devices reduces bandwidth and enhances privacy, with local data storage requirements.

These approaches optimise computational resources, improve real-time performance, and maintain high accuracy in CPS applications like object detection, tracking, and decision-making.

7.1.6. Adaptation Mechanisms

In dynamic and unpredictable environments, CPS models need to be adaptable to changing conditions. Key adaptation mechanisms include:

- Data-driven Adaptation: Utilizes real-world data to enable models to adjust to varying conditions, such as lighting, material properties, and geometric complexities. Techniques like data

 doi:10.20944/preprints202501.0521.v2

augmentation (geometric and photometric transformations) allow models to adapt to unseen scenarios and improve generalization across diverse environments.

- Online Learning: Models are continuously updated with new data collected during deployment. This approach is critical for real-time adjustments to environmental changes, such as varying lighting or evolving cybersecurity threats. Examples include combining offline and online learning techniques, such as the Truncated Gradient Confidence-weighted model, for tasks like image classification.
- Transfer Learning: Involves fine-tuning pre-trained models on smaller task-specific datasets, enabling adaptation to new environments without training from scratch. This is especially useful in CPS where models trained in one context are adapted to others. For example, transfer learning is used to detect attacks in CPS using a ResNet, adapting the model to different operational conditions.
- Ensemble Methods: Combines predictions from multiple models to improve accuracy and robustness. By leveraging diverse architectures (e.g., MobileNetV2, ResNet50), ensemble methods enhance the model's ability to adapt to variations in data quality and features, as seen in medical image analysis tasks where data can differ in noise, resolution, and conditions.
- Adversarial Training: Enhances model resilience to small, intentional perturbations by incorporating adversarial examples into the training process. This helps models adapt to a broader range of input variations, making them more robust against adversarial attacks. Randomized smoothing further strengthens this by adding noise to the input, ensuring the model is resistant to adversarial modifications.
- Federated Learning: Allows distributed CPS devices to train models locally and share updates without centralising sensitive data. This improves performance while maintaining privacy. FL employs various aggregation methods (e.g. averaging, FedGKT) and privacy-preserving techniques (e.g. differential privacy, homomorphic encryption). Challenges include high communication overhead and non-IID data issues, but methods like Fed-MPS reduce resource constraints and communication costs.

These adaptation strategies ensure that CPS models can maintain high performance and resilience in dynamic, real-world environments by continuously learning, adapting, and optimising based on changing conditions.

7.2. Current Advancements and Future Directions

The integration of ML for computer vision into CPS is revolutionizing industries by enabling smarter, more adaptable, and efficient operations. Techniques such as edge AI, self-adaptive RL, and hybrid models exemplify the advancements in CPS, addressing challenges in real-time processing, scalability, and decision-making. These approaches enable applications like healthcare, autonomous vehicles, and smart cities to function more effectively. However, significant challenges remain in scalability, adaptability, and deployment, underscoring the importance of continued research and interdisciplinary collaboration.

Edge AI enhances CPS by performing ML tasks directly at the network edge, allowing for real-time decision-making with reduced latency and improved privacy. By processing data locally, it minimizes reliance on centralized servers, making it ideal for applications requiring rapid responses, such as industrial automation and autonomous systems. Self-adaptive RL contributes by equipping CPS with the ability to adapt to dynamic environments through learning from interactions rather than static datasets. This capability is particularly valuable in robotics and IoT healthcare, where real-time decision-making and resource optimisation are critical. Hybrid models further advance CPS by combining deep learning's feature extraction capabilities with the interpretability of traditional algorithms, creating efficient and scalable solutions for complex tasks like language translation and anomaly detection.

 doi:10.20944/preprints202501.0521.v2

Despite these advancements, significant challenges persist. Scalability issues arise as many synchronization methods assume abundant resources, which are not always available in real-world CPS. Lightweight algorithms that balance accuracy, energy efficiency, and computational costs are urgently needed. Additionally, CPS systems must adapt to dynamic conditions such as environmental variability and real-time constraints, challenges that current methods are often ill-equipped to handle. Technologies like neuromorphic computing and event-based vision require new synchronization strategies to unlock their potential effectively. Continued research is essential to address these gaps, developing innovative algorithms and techniques that ensure CPS systems can meet the demands of diverse and evolving applications.

Interdisciplinary collaboration plays a crucial role in advancing CPS technologies. By bringing together expertise from fields like computer science, engineering, and domain-specific disciplines, researchers can develop solutions that are both technically sound and practically viable. For example, collaboration is essential in integrating emerging technologies such as FL and edge computing with CPS. Edge computing enhances efficiency through strategies like model optimisation, workload partitioning, and real-time monitoring, while FL supports decentralized model training, preserving privacy and enabling robust collaboration across devices. Together, these approaches ensure that CPS systems are not only innovative but also resilient, secure, and scalable.

Ethical and regulatory considerations further highlight the importance of interdisciplinary efforts. Establishing standardized benchmarks fosters consistent evaluation and collaboration. Sustainability and inclusivity must also be prioritized to address environmental concerns and broaden access to these transformative technologies. By combining technical innovation with interdisciplinary collaboration and ethical practices, CPS can continue to evolve, delivering scalable, secure, and responsible solutions that meet the needs of modern industries and society.

References

1. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*. https://doi.org/https://doi.org/10.1136/bmj.n71.
2. Ghanem, M.C.; Mulvihill, P.; Ouazzane, K.; Djemai, R.; Dunsin, D. D2WFP: a novel protocol for forensically identifying, extracting, and analysing deep and dark web browsing activities. *Journal of Cybersecurity and Privacy* **2023**, *3*, 808–829.
3. Dunsin, D.; Ghanem, M.C.; Ouazzane, K.; Vassilev, V. Reinforcement learning for an efficient and effective malware investigation during cyber Incident response. *High-Confidence Computing* **2025**, p. 100299.
4. Ghanem, M.C.; Chen, T.M.; Ferrag, M.A.; Kettouche, M.E. ESASCF: expertise extraction, generalization and reply framework for optimized automation of network security compliance. *IEEE Access* **2023**, *11*, 129840–129853.
5. Ghanem, M.C.; Ratnayake, D.N. Enhancing WPA2-PSK four-way handshaking after re-authentication to deal with de-authentication followed by brute-force attack a novel re-authentication protocol. In Proceedings of the 2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA). IEEE, 2016, pp. 1–7.
6. Ghanem, M.; Mouloudi, A.; Mourchid, M. Towards a scientific research based on semantic web. *Procedia Computer Science* **2015**, *73*, 328–335.
7. Phung, V.H.; Rhee, E.J. A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets. *Applied Sciences* **2019**, *9*. https://doi.org/https://doi.org/10.3390/app9214500.
8. Mirzaei, S.; Kang, J.L.; Chu, K.Y. A comparative study on long short-term memory and gated recurrent unit neural networks in fault diagnosis for chemical processes using visualization. *Journal of the Taiwan Institute of Chemical Engineers* **2022**, *130*. https://doi.org/https://doi.org/10.1016/j.jtice.2021.08.016.
9. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *ArXiv* **2020**, *abs/2010.11929*.
10. Esterle, L.; Grosu, R. Cyber-physical systems: challenge of the 21st century. *Elektrotech. Inftech* **2016**, pp. 299–303.

11. Jahromi Babak Shahian, T.T.; Sabri, C. Real-Time Hybrid Multi-Sensor Fusion Framework for Perception in Autonomous Vehicles. *Sensors* **2018**, *20*, 4357.
12. Bennett, T.R.; Gans, N.; Jafari, R. A data-driven synchronization technique for cyber-physical systems. In Proceedings of the Proceedings of the Second International Workshop on the Swarm at the Edge of the Cloud, New York, NY, USA, 2015; SWEC '15, p. 49–54. https://doi.org/https://doi.org/10.1145/2756755.2756763.
13. Wu, H.; Zeng, L.; Chen, M.; Wang, T.; He, C.; Xiao, H.; Luo, S. Weak surface defect detection for production-line plastic bottles with multi-view imaging system and LFF YOLO. *Optics and Lasers in Engineering* **2024**, *181*, 108369. https://doi.org/https://doi.org/10.1016/j.optlaseng.2024.108369.
14. Robyns, S.; Heerwegh, W.; Weckx, S. A Digital Twin of an Off Highway Vehicle based on a Low Cost Camera. *Procedia Computer Science* **2024**, *232*, 2366–2375. 5th International Conference on Industry 4.0 and Smart Manufacturing (ISM 2023), https://doi.org/https://doi.org/10.1016/j.procs.2024.02.055.
15. Reddy, G.J.; Sharma, D.S.G. Edge AI in Autonomous Vehicles: Navigating the Road to Safe and Efficient Mobility. *International Journal of Scientific Research in Engineering and Management* **2024**, *08*, 1—13. https://doi.org/https://doi.org/10.55041/IJSREM28427.
16. Maier, G.; Pfaff, F.; Wagner, M.; Pieper, C.; Gruna, R.; Noack, B.; Kruggel-Emden, H.; Längle, T.; Hanebeck, U.D.; Wirtz, S.; et al. Real-time multitarget tracking for sensor-based sorting. *Journal of Real-Time Image Processing* **2019**, *16*, 2261–2272.
17. Wang, S.; Zhang, J.; Wang, P.; Law, J.; Calinescu, R.; Mihaylova, L. A deep learning-enhanced Digital Twin framework for improving safety and reliability in human–robot collaborative manufacturing. *Robotics and Computer-Integrated Manufacturing* **2024**, *85*, 102608. https://doi.org/https://doi.org/10.1016/j.rcim.2023.102608.
18. Lesi, V.; Jakovljevic, Z.; Pajic, M. Synchronization of Distributed Controllers in Cyber-Physical Systems. In Proceedings of the 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2019, pp. 710–717. https://doi.org/https://doi.org/10.1109/ETFA.2019.8869467.
19. Prasad, K.P.S.P. Compressed Mobilenet V3: An Efficient CNN for Resources Constrained Platforms. *Purdue University Graduate School. Thesis.* **2021**.
20. Wang, S.; Zheng, H.; Wen, X.; Shang, F. Distributed High-Performance Computing Methods for Accelerating Deep Learning Training. *Journal of Knowledge Learning and Science Technology* **2024**, *3*. https://doi.org/https://doi.org/10.60087/jklst.v3.n3.p108-126.
21. Han, S.; Mao, H.; Dally, W.J. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. *arXiv* **2016**. https://doi.org/https://doi.org/10.48550/arXiv.1510.00149.
22. Bird, J.J.; Kobylarz, J.; Faria, D.R.; Ekárt, A.; Ribeiro, E.P. Cross-Domain MLP and CNN Transfer Learning for Biological Signal Processing: EEG and EMG. *IEEE Access* **2020**, *8*, 54789–54801. https://doi.org/https://doi.org/10.1109/ACCESS.2020.2979074.
23. Hanhirova, J.; Kämäräinen", T.; Seppälä, S.; Siekkinen, M.; Hirvisalo, V.; Ylä-Jääski, A. Latency and throughput characterization of convolutional neural networks for mobile computer vision. *MMSys '18: Proceedings of the 9th ACM Multimedia Systems Conference* **2018**, pp. 204–215.
24. Shen, Y.; Li, Y.; Liu, Y.; Wang, Y.; Chen, L.; Wang, F.Y. Conditional visibility aware view synthesis via parallel light fields. *Neurocomputing* **2024**, *588*, 127644. https://doi.org/https://doi.org/10.1016/j.neucom.2024.127644.
25. Kaur, P.; Khehra, B.S.; Mavi, E.B.S. Data Augmentation for Object Detection: A Review. In Proceedings of the 2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), 2021, pp. 537–543.
26. Wang, H.; Zhang, H.; Zhu, L.; Wang, Y.; Deng, J. ResADM: A Transfer-Learning-Based Attack Detection Method for Cyber–Physical Systems. *Applied Sciences* **2023**, *13*, 13019.
27. Awan, A.A. What is Online Machine Learning? *datacamp.com/blog/what-is-online-machine-learning* **2023**.
28. Tahir, A.; Saadia, A.; Khan, K.; Gul, A.; Qahmash, A.; Akram, R. Enhancing diagnosis: ensemble deep-learning model for fracture detection using X-ray images. *Clinical Radiology* **2024**, *79*, 1394–1402. https://doi.org/https://doi.org/10.1016/j.crad.2024.08.006.
29. Liang, F.; Wu, B.; Wang, J.; Yu, L.; Li, K.; Zhao, Y.; Misra, I.; Huang, J.B.; Zhang, P.; Vajda, P.; et al. FlowVid: Taming Imperfect Optical Flows for Consistent Video-to-Video Synthesis. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition* **2023**, pp. 8207–8216.
30. Bengar, J.Z.; Gonzalez-Garcia, A.; Villalonga, G.; Raducanu, B.; Aghdam, H.H.; Mozerov, M. Temporal Coherence for Active Learning in Videos. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)* **2019**, pp. 914–923.

31. Dantas, P.V.; da Silva Jr, W.S.; Cordeiro, L.C.; Carvalho, C.b. A comprehensive review of model compression techniques in machine learning. *Application Intelligence* **2024**, *54*, 11804–11844.
32. Cai, E.; Juan, D.C.; Stamoulis, D.; Marculescu, D. NeuralPower: Predict and Deploy Energy-Efficient Convolutional Neural Networks. *arXiv:1710.05420* **2017**.
33. He, K.; Gkioxari, G.; Dollär, P.; Girshick, R. Mask R-CNN. *arxiv.org/1703.06870v3* **2018**.
34. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *Computer Vision - ECCV 2016. Lecture Notes in Computer Science ()* **2016**, *9905*, 21–37.
35. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. *arxiv:2005.12872v3* **2020**.
36. Hu, J.; Yan, C.; Liu, X.; Li, Z.; Ren, C.; Zhang, J.; Peng, D.; Yang, Y. An integrated classification model for incremental learning. *Multimedia Tools and Applications* **2021**, *80*, 17275–17290.
37. Shoukat, M.U.; Yan, L.; Yan, Y.; Zhang, F.; Zhai, Y.; Han, P.; Nawaz, S.A.; Raza, M.A.; Akbar, M.W.; Hussain, A. Autonomous driving test system under hybrid reality: The role of digital twin technology. *Internet of Things* **2024**, *27*, 101301. https://doi.org/https://doi.org/10.1016/j.iot.2024.101301.
38. Pan, Y.; Luo, K.; Liu, Y.; Xu, C.; Liu, Y.; Zhang, L. Mobile edge assisted multi-view light field video system: Prototype design and empirical evaluation. *Future Generation Computer Systems* **2024**, *153*, 154–168. https://doi.org/https://doi.org/10.1016/j.future.2023.11.023.
39. Sakina, M.; Muhammad, I.; Abdullahi, S.S. A multi-factor approach for height estimation of an individual using 2D image. *Procedia Computer Science* **2024**, *231*, 765–770. https://doi.org/https://doi.org/10.1016/j.procs.2023.12.140.
40. Kaushik, H.; Kumar, T.; Bhalla, K. iSecureHome: A deep fusion framework for surveillance of smart homes using real-time emotion recognition. *Applied Soft Computing* **2022**, *122*, 108788. https://doi.org/https://doi.org/10.1016/j.asoc.2022.108788.
41. Murel, J.; Kavlakoglu, E. What is object detection? *ibm.com/topics/object-detection* **2024**.
42. GreeksforGreeks. What is Object Detection in Computer Vision? *geeksforgeeks.org/what-is-object-detection-in-computer-vision/* **2024**.
43. Himeur, Y.; Varlamis, I.; Kheddar, H.; Amira, A.; Atalla, S.; Singh, Y.; Bensaali, F.; Mansoor, W. Federated Learning for Computer Vision. *arXiv:2308.13558v1* **2023**.
44. Sprague, M.R.; Jalalirad, A.; Scavuzzo, M.; Capota, C.; Neun, M.; Do, L.; Kopp, M. Asynchronous Federated Learning for Geospatial Applications. *Communications in Computer and Information Science* **2019**, *967*. https://doi.org/https://doi.org/10.1007/978-3-030-14880-5_2.
45. He, C.; Shah, A.D.; Tang, Z.; Sivashunmugam, D.F.N.; Bhogaraju, K.; Shimpi, M.; Shen, L.; Chu, X.; Soltanolkotabi, M.; Avestimehr, S. FedCV: A Federated Learning Framework for Diverse Computer Vision Tasks. *Computer Vision and Pattern Recognition* **2021**. https://doi.org/https://doi.org/10.48550/arXiv.2111.11066.
46. Snell, J.; Swersky, K.; Zemel, R.S. Prototypical Networks for Few-shot Learning. *arXiv* **2017**. https://doi.org/https://doi.org/10.48550/arXiv.1703.05175.
47. Dey, S.; Dutta, A.; Toledo, J.I.; Ghosh, S.K.; Lladós, J.; Pal, U. SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification. *ArXiv* **2017**, *abs/1707.02131*. https://doi.org/https://doi.org/10.48550/arXiv.1707.02131.
48. Finn, C.; Abbeel, P.; Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. *arXiv* **2017**. https://doi.org/https://doi.org/10.48550/arXiv.1703.03400.
49. Santoro, A.; Bartunov, S.; Botvinick, M.; Wierstra, D.; Lillicrap, T. Meta-Learning with Memory-Augmented Neural Networks. In Proceedings of the Proceedings of The 33rd International Conference on Machine Learning; Balcan, M.F.; Weinberger, K.Q., Eds., New York, New York, USA, 20–22 Jun 2016; Vol. 48, *Proceedings of Machine Learning Research*, pp. 1842–1850.
50. Golovin, D.; Solnik, B.; Moitra, S.; Kochanski, G.; Karro, J.E.; Sculley, D., Eds. *Google Vizier: A Service for Black-Box Optimization*, 2017.
51. Yang, H.s.; Kupferschmidt, B. Time Stamp Synchronization in Video System. *International Telemetering Conference Proceedings* **2010**.
52. Hu, Y.; Liu, S.; Abdelzaher, T.; Wigness, M.; David, P. Real-time task scheduling with image resizing for criticality-based machine perception. *Real-Time Systems* **2022**, *58*, 430–455. https://doi.org/https://doi.org/10.1007/s11241-022-09387-6.
53. Ben-Num, T.; Hoefler, T. Demystifying Parallel and Distributed Deep Learning: An In-depth Concurrency Analysis. *ACM Computing Surveys (CSUR)* **2019**, *52*, 1–43.

54. Hinton, G.; Vinyals, O.; Dean, J. Distilling the Knowledge in a Neural Network. *arXiv* **2015**. https://doi.org/https://doi.org/10.48550/arXiv.1503.02531.
55. Cai, G.; Li, J.; Liu, X.; Chen, Z.; Zhang, H. Learning and Compressing: Low-Rank Matrix Factorization for Deep Neural Network Compression. *Applied Sciences* **2023**, *13*. https://doi.org/https://doi.org/10.3390/app13042704.
56. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**. https://doi.org/https://doi.org/10.48550/arXiv.1704.04861.
57. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2020**. https://doi.org/https://doi.org/10.48550/arXiv.1905.11946.
58. Mumuni, A.; Mumuni, F. Data augmentation: A comprehensive survey of modern approaches. *Array* **2022**, *16*, 100258. https://doi.org/https://doi.org/10.1016/j.array.2022.100258.
59. Deng, S.; Zhao, H.; Fang, W.; Yin, J.; Dustdar, S.; Zomaya, A.Y. Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence. *IEEE Internet of Things Journal* **2020**, *7*, 7457–7469. https://doi.org/https://doi.org/10.1109/JIOT.2020.2984887.
60. Tron, R.; Vidal, R. Distributed computer vision algorithms through distributed averaging. In Proceedings of the CVPR 2011, 2011, pp. 57–63. https://doi.org/https://doi.org/10.1109/CVPR.2011.5995654.
61. Lu, J.; Zhao, P.; Hoi, S.C.H. Online Passive-Aggressive Active learning. *Machine Learning* **2016**, *103*, 141–183.
62. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and Harnessing Adversarial Examples. *arXiv* **2015**. https://doi.org/https://doi.org/10.48550/arXiv.1412.6572.
63. Cohen, J.M.C.; Rosenfeld, E.; Kolter, J.Z. Certified Adversarial Robustness via Randomized Smoothing. *arXiv* **2019**. https://doi.org/https://doi.org/10.48550/arXiv.1902.02918.
64. Jiang, S.; Wang, X.; Que, Y.; Lin, H. Fed-MPS: Federated learning with local differential privacy using model parameter selection for resource-constrained CPS. *Journal of Systems Architecture* **2024**, *150*.
65. Boldo, M.; De Marchi, M.; Martini, E.; Aldegheri, S.; Quaglia, D.; Fummi, F.; Bombieri, N. Real-time multi-camera 3D human pose estimation at the edge for industrial applications. *Expert Systems with Applications* **2024**, *252*, 124089. https://doi.org/https://doi.org/10.1016/j.eswa.2024.124089.
66. Rehman, M.; Petrillo, A.; Forcina, A.; Felice, F.D. Metaverse Simulator for Emotional Understanding. *Procedia Computer Science* **2024**, *232*, 3216–3228.
67. Lv, J.; Li, X.; Sun, Y.; Zheng, Y.; Bao, J. A bio-inspired LIDA cognitive-based Digital Twin architecture for unmanned maintenance of machine tools. *Robotics and Computer–Integrated Manufacturing* **2023**, p. 102489. https://doi.org/https://doi.org/10.1016/j.rcim.2022.102489.
68. García-Valls, M.; Chirivella-Ciruelos, A.M. CoTwin: Collaborative improvement of digital twins enabled by blockchain. *Future Generation Computer Systems* **2024**, *157*, 408–421. https://doi.org/https://doi.org/10.1016/j.future.2024.03.044.
69. Piardi, L.; Leitão, P.; Queiroz, J.; Pontes, J. Role of digital technologies to enhance the human integration in industrial cyber–physical systems. *Annual Reviews in Control* **2024**, *57*, 100934. https://doi.org/https://doi.org/10.1016/j.arcontrol.2024.100934.
70. Wang, B.; Zheng, P.; Yin, Y.; Shih, A.; Wang, L. Toward human-centric smart manufacturing: A human-cyber-physical systems (HCPS) perspective. *Journal of Manufacturing Systems* **2022**, *63*, 471–490. https://doi.org/https://doi.org/10.1016/j.jmsy.2022.05.005.
71. Xia, L.; Lu, J.; Lu, Y.; Hao, Z.; Fan, Y.; Zhang, Z. Augmented reality and indoor positioning based mobile production monitoring system to support workers with human-in-the-loop. *Robotics and Computer-Integrated Manufacturing* **2024**, *86*, 102664. https://doi.org/https://doi.org/10.1016/j.rcim.2023.102664.
72. Xiao, W.; Li, A.; Cassandras, C.G.; Belta, C. Toward model-free safety-critical control with humans in the loop. *Annual Reviews in Control* **2024**, *57*, 100944. https://doi.org/https://doi.org/10.1016/j.arcontrol.2024.100944.
73. Tomelleri, F.; Sbaragli, A.; Piacariello, F.; Pilati, F. Safe Assembly in Industry 5.0: Digital Architecture for the Ergonomic Assembly Worksheet. *Procedia CIRP* **2024**, *127*, 68–73.
74. Yedilkhan, D.; Kyzyrkanov, A.E.; Kutpanova, Z.A.; Aljawarneh, S.; Atanov, S.K. Intelligent obstacle avoidance algorithm for safe urban monitoring with autonomous mobile drones. *Journal of Electronic Science and Technology* **2024**, *22*, 100277. https://doi.org/https://doi.org/10.1016/j.jnlest.2024.100277.
75. Meuser, T.; Lovén, L.; Bhuyan, M.; Patil, S.G.; Dustdar, S.; Aral, A.; Bayhan, S.; Becker, C.; Lara, E.d.; Ding, A.Y.; et al. Revisiting Edge AI: Opportunities and Challenges. *IEEE Internet Computing* **2024**, *28*, 49—59. https://doi.org/https://doi.org/10.1109/MIC.2024.3383758.

76. Rocha, A.; Monteiro, M.; Mattos, C.; Dias, M.; Soares, J.; Magalhães, R.; Macedo, J. Edge AI for Internet of Medical Things: A literature review. *Computers and Electrical Engineering* **2024**, *116*, 109202. https://doi.org/https://doi.org/10.1016/j.compeleceng.2024.109202.
77. Feit, F.; Metzger, A.; Pohl, K. Explaining Online Reinforcement Learning Decisions of Self-Adaptive Systems. *arXiv* **2022**. https://doi.org/https://doi.org/10.48550/arXiv.2210.05931.
78. Zhou, X.; Yang, J.; Li, Y.; Li, S.; Su, Z. Deep reinforcement learning-based resource scheduling for energy optimization and load balancing in SDN-driven edge computing. *Computer Communications* **2024**, *226-227*, 107925. https://doi.org/https://doi.org/10.1016/j.comcom.2024.107925.
79. Singh, R.; Bengani, V. Hybrid Learning Systems: Integrating Traditional Machine Learning With Deep Learning Techniques. *ResearchGate* **2024**. https://doi.org/https://doi.org/10.13140/RG.2.2.34709.54248/1.
80. Kadhim, Y.A.; Guzel, M.S.; Mishra, A. A Novel Hybrid Machine Learning-Based System Using Deep Learning Techniques and Meta-Heuristic Algorithms for Various Medical Datatypes Classification. *Diagnostics* **2024**, *14*. https://doi.org/https://doi.org/10.3390/diagnostics14141469.
81. Jia, J.; Liang, W.; Liang, Y. A Review of Hybrid and Ensemble in Deep Learning for Natural Language Processing. *arXiv* **2023**. https://doi.org/https://doi.org/10.48550/arXiv.2312.05589.
82. Abobeah, R.; Shoukry, A.; Katto, J. Video Alignment Using Bi-Directional Attention Flow in a Multi-Stage Learning Model. *IEEE Access* **2020**, *8*, 18097–18109. https://doi.org/https://doi.org/10.1109/ACCESS.2020.2967750.
83. Gautier, T.; Le Guernic, P.; Besnard, L.; Talpin, J.P. The polychronous model of computation and Kahn process networks. *Science of Computer Programming* **2023**, *228*, 102958. https://doi.org/https://doi.org/10.1016/j.scico.2023.102958.
84. Becker, P.H.E.; Arnau, J.M.; González, A. Demystifying Power and Performance Bottlenecks in Autonomous Driving Systems. In Proceedings of the 2020 IEEE International Symposium on Workload Characterization (IISWC), 2020, pp. 205–215. https://doi.org/https://doi.org/10.1109/IISWC50251.2020.00028.
85. Benyahya, M.; Collen, A.; Nijdam, N.A. Analyses on standards and regulations for connected and automated vehicles: Identifying the certifications roadmap. *Transportation Engineering* **2023**, *14*, 100205. https://doi.org/https://doi.org/10.1016/j.treng.2023.100205.
86. Wolford, B. What is GDPR, the EU's new data protection law? https://gdpr.eu/what-is-gdpr/. Accessed: 2025-01-09.
87. Farzaan, M.A.M.; Ghanem, M.C.; El-Hajjar, A.; Ratnayake, D.N. Ai-enabled system for efficient and effective cyber incident detection and response in cloud environments. *arXiv preprint arXiv:2404.05602* **2024**.
88. Wang, X.; Zhu, W. Advances in neural architecture search. *National Science Review* **2023**, *11*. https://doi.org/https://doi.org/10.1093/nsr/nwae282.
89. Avval, S.S.P.; Eskue, N.D.; Groves, R.M.; Yaghoubi, V. Systematic review on neural architecture search. *Artificial Intelligence Review* **2025**, *58*. https://doi.org/https://doi.org/10.1007/s10462-024-11058-w.
90. Chandrala, J. Transfer Learning: Leveraging Pre-trained Models for New Tasks. *International Journal of Research and Analytical Reviews* **2017**, *4*, 809–815.
91. Liu, S.S. Unified Transfer Learning Models in High-Dimensional Linear Regression. *arXiv* **2024**.
92. He, Z.; Sun, Y.; Liu, J.; Li, R. TransFusion: Covariate-Shift Robust Transfer Learning for High-Dimensional Regression. *arXiv* **2024**.
93. He, Z.; Sun, Y.; Liu, J.; Li, R. AdaTrans: Feature-wise and Sample-wise Adaptive Transfer Learning for High-dimensional Regression. *arXiv* **2024**.
94. Lee, J.H.; Kvinge, H.J.; Howland, S.; New, Z.; Buckheit, J.; Phillips, L.A.; Skomski, E.; Hibler, J.; Corley, C.D.; Hodas, N.O. Adaptive Transfer Learning: a simple but effective transfer learning, 2021.
95. Guo, W.; Zhuang, F.; Zhang, X.; Tong, Y.; Dong, J. A comprehensive survey of federated transfer learning: challenges, methods and applications. *Frontiers of Computer Science* **2024**, *18*. https://doi.org/https://doi.org/10.1007/s11704-024-40065-x.