A Self-Adaptive CPS-Based Object Recognition Framework for Smart Glasses Using Dist-YOLOv3-Xception with Attention

Aradea[1*], Irfan Darmawan[2], Rianto[1], Husni Mubarok[1], Ghatan Fauzi Nugraha[1]
[1] Department of Informatics, Faculty of Engineering, University of Siliwangi, Indonesia
[2] Department of Information System, Telkom University, Bandung, Indonesia
E-mail: aradea@unsil.ac.id, irfandarmawan@telkomuniversity.ac.id, rianto@unsil.ac.id, husni.mubarok@unsil.ac.id, Ghatan.Fauzi.Nugraha@unj.ac.id
*Coresponding Author

Keywords: cyber-physical systems, dist-yolov3, light intensity enhancement, pattern recognition, self-adaptation.

Received: April 10, 2025

The Industrial Era 4.0 has emerged as a response to the changes occurring in the world in a dynamic, unexpected, and uncertain manner. This situation requires analytical, predictive, and adaptive capabilities in an intelligent environment. This affects real-world object recognition in vision systems, which are frequently limited to specific signals. Thereby, it creates an adaptive gap. One potential solution to this problem is the development of self-adaptive cyber-physical systems (hereafter, SACPS) to enhance adaptability in recognizing diverse real-world objects. This paper introduces the SACPS model through an extended machine learning/deep learning model applied to smart glasses, which can detect and calculate object distances adaptively. The components of the developed model comprise smart glasses, contextual knowledge, and adaptive requirements based on the SACPS concept. We developed a pre-trained model by combining the Dist-YOLOv3 algorithm with Xception and an attention layer to obtain more optimal results. This research compared the new pre-trained model with those from previous research. Based on the evaluation, the model demonstrates improved performance compared to the baseline when tested on the KITTI dataset, recording a mean Recall (mRec) of 45.21%, mean Precision (mPrec) of 14.73%, and mean Average Precision (mAP) of 30.04%. Additionally, the adaptive system's response to increasing light intensity below 50 revealed good stability, with average post-enhancement brightness reaching 100.0703 (pixel intensity scale). These results demonstrate the significant potential of our model in handling changing environments with strong adaptation in diverse real-world object recognition scenarios. In the case of smart glass, the employment of SACPS can provide good adaptability in predicting distance and increasing light intensity.

Povzetek: Predstavljen je samo-adaptivni CPS-model za pametna očala z izboljšanim Dist-YOLOv3, kjer Xception in pozornostna plast izboljšata prepoznavanje objektov in razdalj. Model doseže višji mAP (30.04 %) ter stabilno prilagajanje pri nizki svetlosti (povprečna osvetlitev 100.07).

1 Introduction

Scholars have declared that the industry 4.0 era is characterized by volatility, uncertainty, complexity, and ambiguity. In particular, it accentuates the state of the world, characterized by rapid change, inadequate predictability, the absence of a cause-and-effect chain, and the blurriness of reality. Another problem arises when providing a real-world object recognition system that incorporates complexity based on various gestures and related devices**.** This can pose challenges for developers. Additionally, they should ensure that the system can recognize a wide range of real-world objects. As a result, the target is to enable the system to learn from the recognition process it carries out and develop its learning process to recognize every gesture and device in the real world. However, the predominant challenge of this situation is how to respond creatively and employ adaptive strategies to face the future [1]. Regarding this issue, Cyber-physical systems (CPS) have emerged as one of the latest advancements in this era. CPS has been a trending topic among academics and practitioners [2], [3]. In its various applications in the real world, CPS can be utilized to meet distinct system requirements in the Industry 4.0 era [4]. For example, CPS can integrate the virtual and physical worlds, meeting the predominant characteristics of Industry 4.0 requirements. However, the operating environment can vary and cover distinct uncertainties [5]. In this matter, a pivotal feature of the present and future breakthrough is self-adaptive systems (SAS) [3]. AS is a system that can modify its behaviors based on changes occurring either in the environment or within the system itself. Unfortunately, several gaps remain regarding the implementation of SAS in CPS, including insufficient information on the characteristics of SAS, particularly those related to CPS [3] Later, in the development process, not all software engineering knowledge can be implemented [6], Furthermore, not all software engineering knowledge can be effectively implemented during the development process [6]. Architectural

requirements should center on adaptation to inform appropriate architectural decisions. Antonino et al. [2] argue that current standard software development approaches are unable to represent complex contexts. Hence, they have not been able to introduce fatal complexity to CPS. Therefore, a contextual modeling approach is required, namely, modeling system entities with specific contextual attributes. In particular, SAS in CPS requires a dynamic context. This occurs because uncertainty prevents the system from knowing its current state. An approach to documenting uncertainty that integrates other artifacts from different perspectives is required to specifically capture uncertainty [7].

The CPS design should be able to resolve uncertainty at runtime. Consequently, SAS should be a fundamental approach for the system to meet its functional and performance specifications [8]. Zavala E. et al. [5] argue for the need for distributed runtime models in CPS to capture operational state and context as a form of knowledge representation. In this case, the runtime model is typically implemented through the MAPE-K control loop, which combines new knowledge through decentralized operations. Thus, it allows for conflicts [9]. Based on this description, it is necessary to develop a flexible knowledge structure with a reasoning mechanism at run-time.

Likewise, CPS presents diverse issues in terms of system design, implementation, and maintenance. One of the main issues is the need for adaptive techniques to cope with a dynamic and constantly changing environment [3]. CPS necessitates a SAS that can monitor and adjust its behavior based on changes in its environment [4]. Antonino et al. [2] have identified significant required specifications directly from the adaptive requirements architecture for CPS and enabled IoT. Conversely, the implementation of SAS in CPS raises issues regarding interoperability, integration, and technical assessment in the system [7].

This paper is aimed at developing a generic model for adaptive service in CPS to recognize real-world objects. Combining formalized approaches with the CPS system metamodel provides the possibility to invigorate semantic interoperability. Moreover, it can enhance its performance [8]. The system framework functions to capture and handle a variety of CPS variability. As a result, developers can produce products that can be applied to modern software system environments for the needs of diversified domains based on the demands of the current world. The cultivated strategy is to expand the CPS architecture by embedding the SAS approach through adapting machine learning/deep learning methods. Technically, it modifies the Dist-YOLOv3 [10] algorithm by substituting the original architecture of YOLOv3, namely Darknet53 with the Xception architecture [11] added with an attention mechanism layer. The proposed model is represented as a generic knowledge structure to accommodate the need for recognizing various real-world resources and objects. The adaptability mechanism is determined through a learning model that can capture instances or concrete CPS services based on contextual requirements operationalizing at run-time.

This model is implemented in the case of smart glasses, namely combining concepts from object recognition, SAS, and CPS into one unified whole. The utilization of object recognition was selected since it can better comprehend the circumstances and conditions of the surrounding environment compared to a sensor [12]. SAS and CPS provide smart glasses with flexibility in interacting with users and high adaptability while encountering environments with low light intensity. By doing so, this study offers a significant impact on the progress of assistive devices for the visually impaired. The key contributions of this study are as follows:

1. A novel CPS-based object recognition architecture for smart glasses that integrates self-adaptive capabilities to handle environmental uncertainty, particularly in low-light conditions.
2. An enhanced version of Dist-YOLOv3, by replacing the Darknet53 backbone with the Xception architecture and integrating an attention mechanism to improve object detection accuracy and distance estimation.
3. An adaptive light intensity enhancement module enables the system to dynamically adjust image brightness, thereby improving detection performance under poor lighting conditions.
4. Empirical validation on the KITTI dataset, demonstrating improved performance in terms of mean Average Precision (mAP), recall, and precision compared to the original Dist-YOLOv3 model
5. A unified smart glasses framework for visually impaired individuals, capable of real-time object recognition, distance estimation, and auditory notification using text-to-speech.

The remaining sections of this paper encompass discussing related work (second section), describing the proposed model (the third section), and discussing experiments (the fourth section) (e.g. a discussion of the case study and its evaluation results). Finally, the fifth section infers the entire results of the work and discusses future directions of investigative attempts.

2 Related work

Nowadays, researchers have proposed assorted approaches to address emerging challenges. Grounded in the investigative results of the most current survey and technical papers, there have been several requirements that Self-Adaptive Cyber-Physical Systems should possess. As an example, studies discussing system modeling, system evolution, supporting contextual uncertainty, and system evolution requirements were conducted by Zavala et al. [5], Antonino et al. [2], Petrovska et al. [4], Jehn-Ruey Jiang [13] and Habib et al [14]. Scrutiny emphasizing system learning and adaptation and handling contextual uncertainty was performed by Zhou et al.[15], Búr et al. [16], Weyns et al [17], Ahmed et al. [18], and Sony [19]. Investigative efforts focusing on syntactic and semantic interoperability, system collaboration, and integration between physical and virtual systems, including handling

contextual uncertainty at run-time were performed by Kluge [20], Weichhart et al. [21], Casadei et al. [22] and Aradea et al [23], [24].

Based on a description of reviewed prior scrutiny, Table 1 designates a comparison of related works specified into design-time and run-time requirements, including their strengths and weaknesses.

Table 1: Comparison of related works

Works	Design-time specifications	Run-time Specifications
Jehn-Ruey Jiang (2018)	ISA-95 architecture, 5C architecture	8C Architecture
Zavala et.al. (2018)	Contextual model, modeling reference architecture, hierarchical inter-intra-collaborative	Feedback control loop: centralized and decentralized control loops, machine learning
Zhou, et. al. (2018)	Matching network architecture with a non-parametric differential KNN-like classifier	MAML (Model-Agnostic Meta-Learning)
Antonino et.al. (2018)	Adaptation model terms: adaptation context, adaptation stimulus, realization	MAPE-K reference: stimulus, preconditions, postconditions, invariants
Petrovska et.al. (2019, 2020)	Model knowledge (multi-agent), observation aggregation (run-time context), subjective logic (dempster-shafer)	MAPE-K loops *(*master-slave): subjective opinion creator, knowledge aggregator, cumulative belief fusion, cumulative belief fusion
Kluge (2020)	Model-driven: graph rewriting rules, role-based context-model: CPS, decentral adaptations: decentral role-based system	MAPE-K loops: distributed role runtime: communicating sequential processes, adaptation plan: role instance model
Búr et. al. (2020)	Model run-time: monitoring rules, execution planner-optimizer, code generator, distributed graph queries	Computing platform: distributed runtime monitoring, model update operations, local search-based pattern matching
Bandyszak et.al. (2020)	Model-based approach: modeling behavioral requirements (structural operational context), context uncertainty	Ontological: orthogonal uncertainty models, system context & and requirements ontology, uncertainty ontology
Sony (2020)	8C Architecture	Lean Six Sigma (LSS)
Weichhart, et. al. (2021)	Orchestration software model: ad-hoc planning, re-planning, BPMN language, NgMPPS Engine, syntactic, semantic & pragmatic modeling	Interoperability run-time model: semantic interoperability, pragmatic interoperability, REST web service
Weyns, et.al. (2021)	Crossing Boundaries, Leveraging the Human, Fluid Modelling, On the Fly Coalitions	Dynamically Assured Resilience, Learn Novel Tasks
Casade, et.al. (2021)	Augmented Collective Digital Twins: holistic, declarative, and integrated system view.	Integrating physical and virtual devices and meta-models for self-organizing
Habib, et.al. (2022)	IIRA (Industrial Internet Reference Architecture), RAMI 4.0 (Reference Architecture Model Industrie 4.0)	IMSA (Intelligent Manufacturing Systems Architecture), Merge of IEC and ISO standards for smart manufacturing.

Table 1 illustrates assorted relevant literature regarding the requirements for CPS to have adaptability. Based on Table 1, we identified three groups of methods/approaches for dealing with the problem of object recognition models based on CPS. First, methods were adopted to handle contextual uncertainty, such as those applied by [2], [4], [5], [13], [14]. Generally, it indicated advantages in recognizing the context of a CPS environment based on certain contextual requirements. However, it has not yet supported comprehensive CPS domain modeling. Second, focusing on learning mechanisms for system adaptation as proposed by [15]–[17], [19], [34], [35] they offered machine learning approaches dedicated to adaptability based on current learning algorithms. Unfortunately, these approaches also pay less attention to the needs in modeling the CPS domain where the system operates. Third, handling contextual uncertainty as proposed by [20]–[24], this approach is prepared to perform domain modeling including handling CPS contextual uncertainty. However, it has not adopted the learning process optimally. As a result, there is still a need to increase the optimized value of these approaches. Grounded in discussions of related works, investigative gaps remain. It becomes our motivation to propose a model to fill these gaps. Our proposal offers a generic model for the adaptability of CPS services in recognizing real-world objects consisting of the ability to model the CPS domain through the integrated cyber system and physical system architectures influenced

by contextual knowledge. Apart from that, increasing the optimized value of the learning process was executed by expanding the Dist-YOLOv3 architecture by modifying the backbone by embedding the Xception architecture as a replacement for Darknet53. Further, to capture broader real-world cues, we accentuate the mechanism layer between the neck and head of the architecture.

3 Proposed method

The target of our developed generic model can be applied to recognize a variety of real-world objects in the CPS environment. Hence, the system framework should be able to capture and handle various CPS variabilities. The developed strategy is to expand the CPS architecture with learning process capabilities for all instances or concrete CPS services. The adopted approach is SAS through the development of machine learning/deep learning methods as a control process at run-time. The mechanism of the adaptability learning process is determined through the contextual requirements of the CPS environment functioning at run-time. Figure 1 displays our proposed architectural model.

The architecture in Figure 1 is an extension of the architecture proposed by Habib et al. [14] with additional modifications to SAS [23], [24] which we have previously developed, called self-adaptive cyber-physical systems (SACPS). Our previous model formulated an adaptive model based on contextual knowledge through a probabilistic reasoning approach. More specific models can be viewed in papers [23], [24]. In this paper, we have built up the model with several adjustments from machine learning/deep learning methods. The addition of the SAS component is intended to enable the object recognition model to adapt to uncertainty originating from contextual knowledge of a CPS developing environment. Grounded in the architecture of Figure 1, the development of the SACPS-based object recognition model is created into a continuous cycle and the model can continue to self-adapt. By doing so, the system embedded in the model can continue to be updated according to the requirements to handle uncertainty. Specifically, the components of the SACPS architecture that we propose consist of:

a. Smart Connection

Smart connection was the first stage of the cultivating processes of an object recognition model. In the smart connection section, the process of assembling the physical components for the object recognition model was conducted. These components were assembled in such a way that they could be a source of required data collection to inform the conversion process. The applied components should be installed properly and connected to other components. In other words, communication among components and other devices via the internet network could be barrier-free. By this description, the data collection process was carried out in the smart connection section while preparing to continue to the next section.

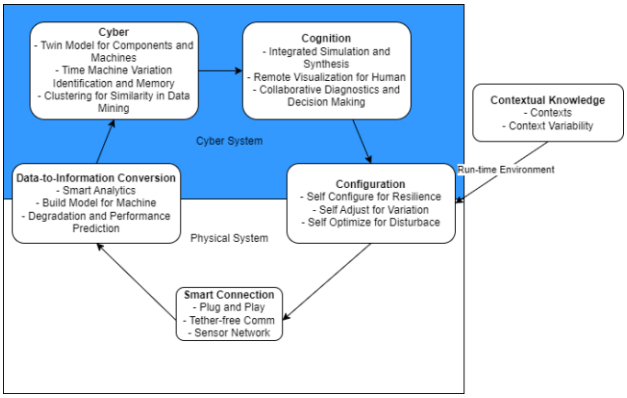


Figure 1: SACPS (self-adaptive cyber-physical system)

b. Data to Information Conversion

This section performed the data conversion process into information read by machines. The conversion process could be executed in miscellaneous ways based on the developed model. In this case, the data conversion process involved the process in Vagjl et al.'s Dist-YOLOv3 algorithm [10]. This represented a training process from a dataset containing images for the object recognition model. The better the conversion process was carried out at this stage, the more self-awareness properties the machine would have.

c. Cyber

This section was involved in collecting some information from each machine connected to the internet network. The obtained information was utilized as evaluative materials to select which machine or model had better performance. With this in mind, the model with the best performance was applied to each machine. Supporting data (e.g. historical data from each machine) were applied to enhance the performance level of the model.

d. Cognition

At this level, the information-collecting process conducted at the previous level was used as suggestions for making decisions for developers. Besides, a simulated process was also performed on the object recognition model embedded in the machine to analyze whether the provided results were appropriate or still required further development.

e. Configuration

This top-level was created to give the machine the ability to self-configure, self-adjust, and self-optimize. The configuration process was operationalized by noticing the results of decisions made from the cognition stage. In addition, the configuration level was influenced by contextual knowledge [23], [24] from outside the CPS environment. Contextual knowledge was employed as a parameter for system uncertainty which may affect the machine configuration process whether it is required to self-configure, self-adjust, or self-optimize.

In this paper, the SACPS model developed by us will be applied to the needs of a smart glasses system. Figure 2 signifies our proposed smart glasses architecture based on SACPS elements as a result of our previous model extension [23], [24]. Each applied tool in this architecture will be connected via an internet network connected to the data center in the cloud. In this paper, the emphasis is situated on creating a model for smart glasses with object detection capabilities based on the Dist-YOLOv3 algorithm modified in such a way. Thus, it can produce an artificial voice output originating from text-to-speech. Dist-YOLOv3 refers to a variation of the YOLOv3 model [25] by widening the prediction vector from three values ($p = (b, c, o)$) to four values ($p = (b, c, o, d)$ [10]. In this case, b is the bounding box coordinate ($b = (x, y, w, h)$), c is the confidence value for each class, o states the confidence value for the detected object, and d is the distance value for the object. Figure 3 showcases how Dist-YOLO determines the approximate value of the object distance.

In addition, Dist-YOLOv3 expands the calculation of loss function values based on YOLOv3 by adding loss values for object distance prediction [10]. Equations (1) and (2) demonstrate the difference in the loss function formula between YOLOv3 and Dist-YOLOv3.

$$l = \sum_{i=0}^{G^w G^h} \sum_{j=0}^{n^a} q_{i,j} [l_1(i,j) + l_2(i,j) + l_3(i,j) + l_4(i,j)] \quad (1)$$

$$l = \sum_{i=0}^{G^w G^h} \sum_{j=0}^{n^a} q_{i,j} [l_1(i,j) + l_2(i,j) + l_3(i,j) + l_5(i,j)] + l_4(i,j) \quad (2)$$

Equation (2) demonstrates the calculation in determining the loss value in Dist-YOLOv3 [10] as an extension of equation (1) encompassing the loss value for YOLOv3 [25] by adding up all the existing components, namely $l_1(i,j)$ refers to the value loss for bounding box center prediction, $l_2(i,j)$ refers the loss value for the box dimension, $l_3(i,j)$ is described as the confidence loss, $l_4(i,j)$ is illustrated as the class prediction loss, and $l_5(i,j)$ is delineated as the loss value for object distance prediction. Scrutiny conducted by Vagjl et al [22] proved that the application of the Dist-YOLOv3 algorithm can provide detecting results for an object equipped with a prediction of the object's distance to the camera.

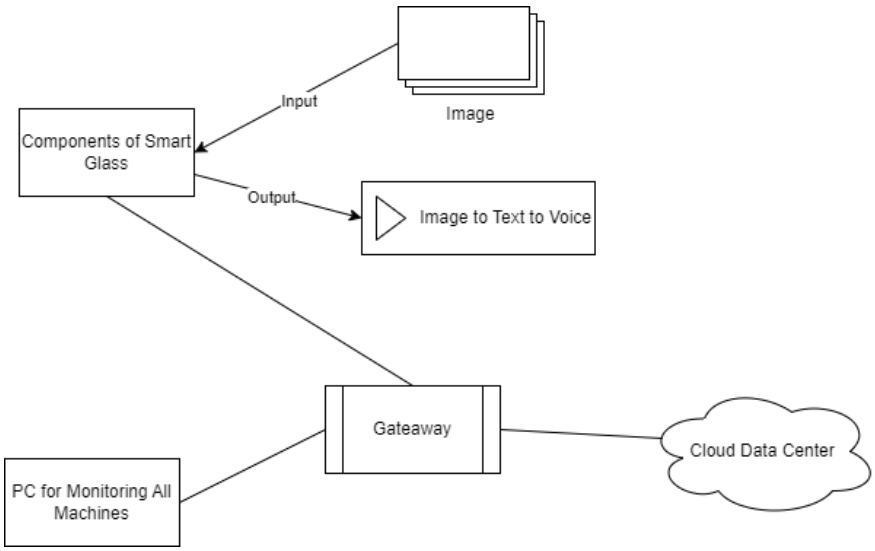


Figure 2: Smart glasses architecture

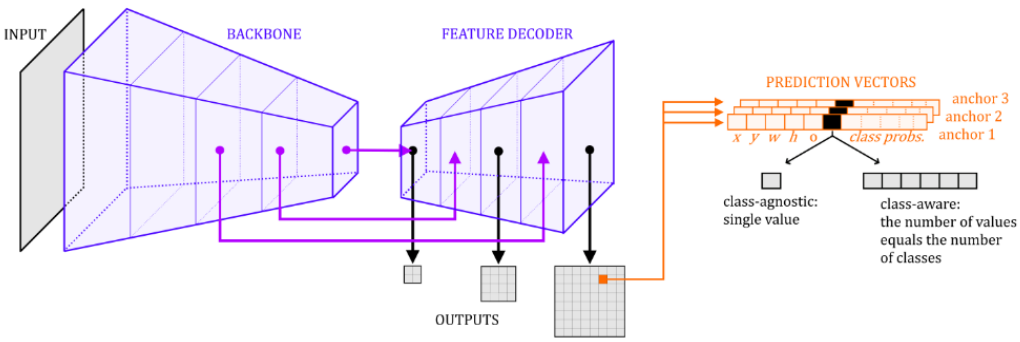


Figure 3: Illustration of Dist-YOLOv3 in determining object distance predictions [10]

Even though it shows a good performance in testing, Dist-YOLOv3 has quite large loss values. Therefore, a modification is needed to overcome this problem. This study offers a solution by applying the Xception architecture [11] combined with the attention mechanism layer [26]. The implementation of the Xception architecture is adopted to replace the YOLOv3 backbone utilizing the Darknet53 architecture. On the other hand, the attention mechanism layer is applied to the head section of the Dist-YOLOv3 architecture aimed at enhancing feature representation. In particular, it focuses attention on more relevant parts of the feature map. Thus, improving the focus on pivotal information to produce more accurate predictions can be realized.

Figure 4 reports the implementation of the Xception architecture and Attention mechanism in the Dist-YOLOv3 approach. The implementation of Xception was placed in the backbone to replace the previous backbone, namely Darknet53. There are three main processes carried out in this architecture, including the entry flow, middle flow, and exit flow stages [11]. The entry flow st age aims at reducing the dimensions of the input image and extracting basic features through three convolutional layers (conv1, conv2, and conv3) and depthwise separable convolutions (sepConv1, sepConv2, sepConv3) [11]. Exiting the entry flow, the next image enters the middle flow section aimed at deepening the network while maintaining feature information through residual connections. The final stage is the exit flow intended to combine the features extracted and process them into the desired outputs. In this case, the output produced is in the form of extracted image results performed at three different resolutions including f1: 13x13x1024, f2: 26x26x512, and f3: 52x52x512.

After the backbone produces the image extraction results in three image resolutions, the three extractions enter the neck section with the same functions and layers as in the Dist-YOLOv3 architecture [10]. Unfortunately, what makes it different is that before proceeding to the head section, each result produced by the neck will first be entered into the attention mechanism layer. In particular, the process at this layer occurs as in the pseudocode in Table 2.

The input feature map $x \in \mathbb{R}^{H\times W\times C}$ is first linearly projected into three matrices: Query $Q = W_q \cdot x$, Key $K = W_k \cdot x$, Value $V = W_v \cdot x$, where $W_q, W_k, W_v \in \mathbb{R}^{C\times d}$ are learnable weights and d is the attention dimension. The attention scores are computed using scaled dot-product attention:

$$A = softmax\left(\frac{Q.K^T}{\sqrt{d_k}}\right) \quad (3)$$

This produces an attention matrix $A \in \mathbb{R}^{(H.W)\times(H.W)}$ which determines how much focus each spatial position should give to others. Finally, the enhanced output is obtained as:

$$x_{enhanced} = A \cdot V \quad (4)$$

This yields a refined feature representation where each position aggregates information from relevant spatial contexts, allowing the network to model global dependencies within the image.

Table 2:
Pseudocode of the attention mechanism process

Attention Mechanism Pattern
Input: Feature Map $x \in \mathbb{R}^{H\times W\times C}$
1. Project x into Query (Q), Key (K), and Value (V):
$Q = W_q \cdot x$
$K = W_k \cdot x$
$V = W_v \cdot x$
2. Compute Scaled Dot-Product Attention Scores:
$A = Softmax\left(Q \cdot \frac{K^T}{\sqrt{d_k}}\right)$
3. Compute Weighted Feature Representation:
$x_{enhanced} = A \cdot V$
Output: Enhanced Feature Map x_enhanced

The x value comes from a process in the neck as a result of combining features from miscellaneous resolutions to provide a rich and informative feature representation. The first stage undertaken in this layer is the Calculation of Attention Weights where the Attention mechanism calculates the weights for each feature element. This can be conducted in multiple ways, namely dot product, scaled dot product, or complex self-attention mechanisms such as those applied in transformers. Next, in the Weighted Sum of Features, the input features will be combined with the calculated weights. This produces a new feature map where important features are given greater weight. On the other hand, less important or noisy features are provided with less weight. Furthermore, the Enhanced Feature Map strengthens the feature map combined with features from a lower resolution using the Concatenate layer.

In the diagram from Figure 4, the Backbone (Xception), Neck (feature decoder with upsampling and convolutional refinement layers), and the Head that produces the final object predictions are distinctly separated. The Attention Mechanism is explicitly positioned after the Neck and before the Head, ensuring spatial refinement of multi-scale feature maps prior to prediction. This updated visual structure aligns with the standard object detection pipelines and reflects the actual implementation logic applied in this study.

By adopting this algorithm, it is expected that object detection will be more representative because it incorporates the distance element into each detection process. The model has been formulated by considering the SACPS cycle. By doing so, the target model can possess the ability to self-configure, self-adjust, and self-optimize. As a complement to the model, we also provide features for monitoring requirements to accommodate the involvement of the developers. This feature enables monitoring of the entire model's activities via the internet.

Given this fact, the process of updating and enhancing the model can be performed currently. Also, it can produce better models in terms of performance than others. Furthermore, the requirement for accessible and uninterrupted data support for both smart glasses and

monitoring devices remains vital. Therefore, in the architecture of Figure 2 it is specified that the data will be stored in a cloud-based data center. With this in mind, smart glasses and monitoring devices can access the data at any time as long as they are connected to the internet.

As a form of proposed model validation, an evaluation was conducted through two scenarios. First, it is operated by evaluating the object recognition model. Second, it is conducted by measuring the quality of adaptation performed by SACPS. Technically, the results of the model evaluation are visualized using two metrics, such as Pascal VOC AP and intensity average. The function of the Pascal VOC AP metric is used to evaluate the Dist-YOLOv3 modified model. On the other hand, the average intensity metric is used to calculate the average light intensity resulting from the adaptive process. By doing so, the entire results of this evaluation can demonstrate the entire quality of research related to the development of SACPS and its implementation process. The proposed model still opens up avenues for future research as a form of refinement of its shortcomings. The remaining shortcomings cover subsequent aspects:

1) A very significant architectural overhaul makes the program implementation process difficult,
2) Distance measurement still uses historical data on the label.
3) The YOLOv3 basic model indicates a large number of parameters causing the object recognition model heavy when operated.
4) The adaptive and recognizing process of objects only supports outdoor areas.

Of these shortcomings, there are several potential improvements, including replacing the basic YOLOv3 model with the latest model, namely adopting a more flexible distance calculating concept, and adding an indoor model to make it more universal.

From a systemic perspective, the proposed architecture has been developed within the context of a generic Cyber-Physical System (CPS) metamodel. Each component in the proposed pipeline corresponds to a specific layer in a 5C-like CPS architecture [13]: the camera and ambient sensors represent the Connection layer; the light intensity adaptation module functions as a preprocessing mechanism in the Conversion layer; the object recognition and distance estimation modules form the core of the Cyber layer, handling perception and decision-making; and the auditory feedback system aligns with the Cognition and Configuration layers by providing real-time responses to the user. Although this paper presents a specific implementation for object recognition in smart glasses, the modular and layered design is intended to be extensible for broader adaptive services in CPS. The system's knowledge structure—encompassing perception, adaptation, and feedback—can be generalized for other CPS applications that require environment awareness and user interaction.

More specifically, the Smart Connection layer is operationalized through the image acquisition system and data transfer mechanisms that enable the collection of visual input from the surrounding environment. The Data-to-Information Conversion layer involves preprocessing steps, such as adaptive brightness enhancement, and the object recognition pipeline that transforms raw input into structured outputs. The Cyber layer consists of deep learning-based inference (Dist-YOLOv3 [10] with Xception [11] and attention [26]) and distance estimation logic, which drive perception and situational awareness. The Cognition layer is activated when critical cues (such as proximity or classification confidence) are detected, allowing the system to assess the contextual relevance of outputs. Finally, the Configuration layer is reflected in the auditory notification system, which adapts responses based on recognition results and environment states. This layered mapping reinforces the integration of the SACPS concept within the operational structure of our adaptive object recognition system.

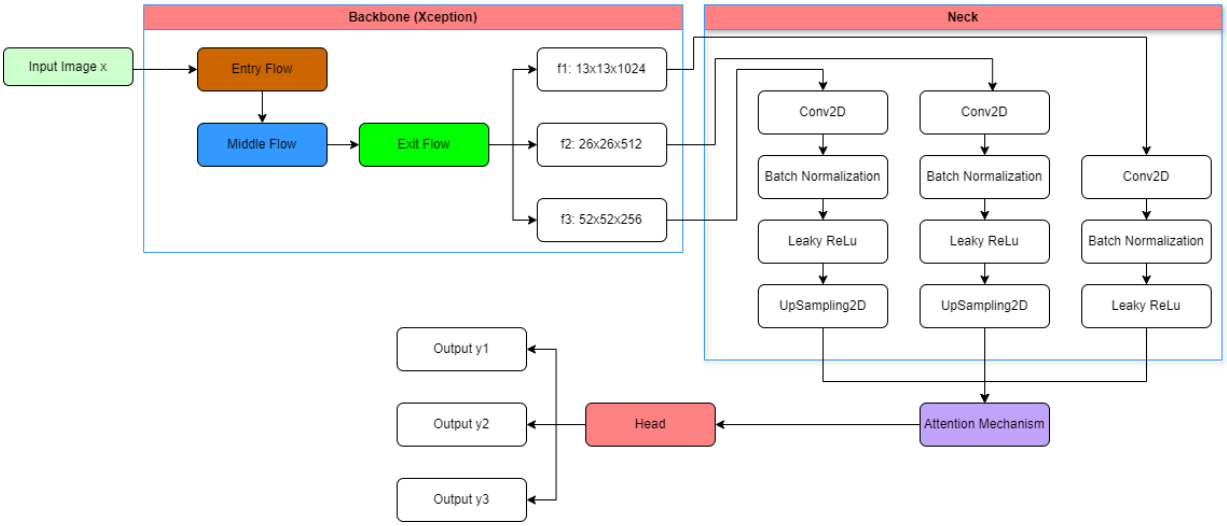


Figure 4: Architectural modifications to Dist-YOLOv3

4 Experiment

The experimental instrument was developed by adopting the guidelines of Wohlin et al. [27] about experimentation in software engineering. Table 3 denotes the overall research design. The elements of aims, object of study, domain, and focus are targets for defining entire indispensable aspects in an experiment. Evaluative questions were a set of questions addressed to characterize how to assess targets and determine the object being measured. On the other hand, variables were metrics or data sets related to each question that should be answered. This investigation aimed at developing a solution to the problematic variability in uncertain real-world objects in a CPS environment characterized by an adaptive strategy to invigorate the object recognition quality. Specifically, the object recognition quality encompassed the ability to identify objects, and distances, and provide sound notifications to users. Further, this study was also intended to evaluate the performing specifications of the object recognition system through measuring loss, validating loss, precision-recall, and average precision (AP) through Pascal VOC AP measurements.

Generally, the experimental process was guided by the domains and elements outlined in Table 3, which include the adaptive strategy and performance specifications of the CPS-based object recognition system. The adaptive strategy was implemented through algorithmic design within the smart glass's architecture, focusing on automatic light intensity adaptation and feature refinement using attention mechanisms. Meanwhile, the performance specification evaluation was conducted using quantitative metrics, including mean Average Precision (mAP), mean Precision (mPrec), mean Recall (mRec), and loss values, to measure the accuracy and reliability of object detection and distance estimation.

Table 3:
Experimental design

No	Elements	Description
1	Aims	a. Developing a CPS-based object recognition model by embedding automatic adaptation (self-adaptation) capabilities to handle uncertainty b. Evaluating the performance of a CPS-based object recognition system
2	Study Objects	a. Adaptive specification of CPS-based object recognition model b. CPS-based object recognition artifact requirements
3	Domain	Smart glasses
4	Foci	a. Adaptive strategy for CPS-based object recognition systems b. Performing specifications for CPS-based object recognition systems
5	Evaluative Questions (PE)	a. PE_1- To what extent can the CPS-based object recognition system maintain accuracy and robustness under uncertain environmental conditions (e.g., low light)? b. PE_2-What is the performing measure of each artifact element of the CPS-based object recognition systems?
6	Variables (V)	a. Response (V_1-system failure; V_2-system functional and non-functional strategies; V_3-new stimulus) b. Measurement (V_4-Pascal VOC AP (Average Precision) evaluation)

This experimental framework also considers the intended application scenario—assistive navigation for visually impaired individuals. Blind individuals face significant challenges in recognizing surrounding objects, and our study aims to address this issue by providing a CPS-based recognition model that delivers real-time feedback. Through the proposed architecture, which includes adaptive brightness enhancement and an optimized deep learning backbone (Xception), the system is designed to enhance recognition accuracy and facilitate safer navigation.

This has become one of the developed applications in the area of self-adaptive cyber-physical systems. The system environment, including users (the visually-impaired people) and all monitoring sources derive from wearable devices. Table 4 signifies the specifications of the CPS cases.

Table 4:
Case specification of cyber-physical system

Components	Specifications
Smart glasses	BM: User Blindness
Context Knowledge	C_1: Object C_2: Object Distance C_3: Low Brightness
Adaptive Requirements	Cv_1: Capturing and predicting an object Cv_2: Even to predict the distance from the object Cv_3: Even with the brightness enhancement Cv_4: Even to release a sound notification

Contextual variability showed uncertainty due to discrete factors, such as unexpected changes, increasing data volumes, inaccurate information, problematic system and service infrastructures, and new and unpredictable situations. Dealing with the case specification in Table 3 (the adaptive process to normal situations), the Cyber-Physical System detected identified objects in the surroundings. Context variability (Cv) can be monitored with $C_1, C_2, C_3 \in \{C_{V1}, C_{V2}, C_{V3}, C_{V4}\}$. Each context value, namely $C_1, C_3 \in \{C_{V3},\ C_{V1}\}$ means the context of the object, and Low brightness was processed to meet the requirements of the event to brightness enhancement and capture and predict the objects. Context $C_1,\ C_2, C_3 \vee C_1, C_2 \in \{C_{V1}, C_{V2}, C_{V3}, C_{V4}\}$ means that the context knowledge obtained will be processed until the system issues a sound notification. In its application, SACPS makes it possible

to deal with a number of situations denoted in the inference model as follows:

Rule-1: if (camera_capture = object) and (object = has_distance) then system_output = give notified the user based on an object for navigation.
Rule-2: if (camera_capture = object) and (object = has_low_brightness) then system_output = increase brightness.
Rule-3: if Rule-2 is True then do Rule-1

Table 5 reveals the algorithm in pseudo-code form for the SACPS mechanism on the smart glass's architecture based on the five existing rules, namely Rule-1 to Rule-3.
Objects (O) would be identified based on context knowledge $C_i \in \{C1, C2, C3\}$, the results of which became a reference for the inference of the formulated model. The process of determining identification was carried out during monitoring (M) with the output in the form of $Cv_i \in \{C_{V1}, C_{V2}, C_{V3}, C_{V4}\}$. The output would be sent to the analyzer_manager section which was at the cognition level (CG).

Table 5:
Sacps adaptive algorithm for smart glasses

Adaptation of CPS Pattern

```
Input
O ← C1, C2, C3
Do
Let
O ← inference model
// Monitoring (M)
For O in  runtime artifact, do
   O ← get value C1, C2 and C3 in runtime artifact
   For each values C1, C2 and C3 in SACPS artifact, do
       If Cvi in runtime artifact, then
           Send information Cvi to analyzer_manager
       endif
   endfor
endfor

// Cognition (CG)
For each Cvi in analyzer_manager, do
   If Cv1, Cv2, Cv4 is True, then
      If Cv1 is True, then
         Increase brightness from O
      Endif
      O ← predict_object and predict_distance
      new_decision ← O
   else
       O ← the object cannot be detected
       new_decision ← empty
   endif
endfor

// Configuration (CF)
If CG contain new_decision, then
   System ← release a sound notification
   For each system in runtime artifact, do
       Send information to M
   endfor
endif
```

Next, Cv_i in analyzer_manager was processed in the Cognition (CG) section to obtain a new decision from the results obtained in M. This process was executed by analyzing Cv_i which would produce a new decision O. The output from CG was in the form of a decision command for the configuration (CF) section. If new_decision contains a new decision, the system will issue a notification sound as the final result of the process of object detection.
The model development process was conducted based on an extension of the machine learning/deep learning method of John et al. [28] adapted to the requirements of this investigation. Hence, the development stages had been determined as represented in Figure 5.

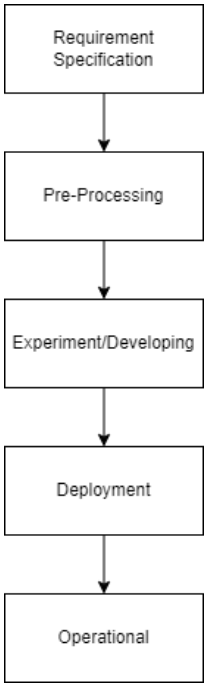


Figure 5: Development stages

The experimental process requires an investigative environment to meet every experimental need while being performed. Therefore, it is necessary to prepare appropriate specifications to fulfill experimental needs. Hence, the obtained results are maximal. The required specifications for the training process are adjusted to the requirements of the Dist-YOLOv3 algorithm Vagjl et al. [10] referring to the specifications of the employed device. The required specification consists of (a) Python 3.9.6; (b) Tensorflow 2.6.0; and (c) CUDA. The training process was conducted using a batch size of 2 and a learning rate of 0.001. The Adam optimizer was used, and training was performed for a maximum of 40 epochs with early stopping based on validation mAP. The loss function combined cross-entropy loss for object classification and smooth L1 loss for distance estimation. The Xception backbone was initialized with pretrained ImageNet weights to leverage transfer learning. Additionally, a single-head self-attention block with an embedding

dimension of 256 was integrated into the detection head. Query, key, and value matrices were generated via 1×1 convo lutions, followed by layer normalization and ReLU activation for feature refinement.

The utilized data for the model creation process originated from a dataset created by KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) Geiger et al. [25]. The dataset was called KITTI 3D Object Detection Evaluation 2017. It refers to a data set covering 3D objects. Also, it consisted of 7481 training data and 7518 test data. The data pre-processing process was executed by following the steps proposed by Vajgl et al. [10] as manifested in Figure 6. The data were separated into two parts, namely training data and test data obtained from the training data section of the KITTI Dataset. This occurred due to only the data in the training section containing distance values. As a result, the actual distance was required to be compared with the predicted distance from the detection results for testing purposes. The emphasis of preprocessing was on label processing containing prominent information for the training process. Information (e.g. bounding box points, class index, and distance value) were elements that should be included in label annotations.

In the experimental stage, the training process was carried out by making adjustments to the employed architecture. The Darknet53 architecture was substituted by the Xception architecture [11] with the addition of an attention mechanism layer proven to improve the performance of the applied architecture in YOLO [26], [27], [28]. In proving that the created model is better than the model of Vajgl et al. [10], we compared the pre-trained model from this study with the pre-trained model available at https://gitlab.com/EnginCZ/yolo-with-distance established by Vajgl et al. [10]. This comparison aims to compare state-of-the-art methods discussed in the related work section, particularly the work of Vajgl et al. [10]. Figure 7 illustrates the difference in train loss and validation loss values.

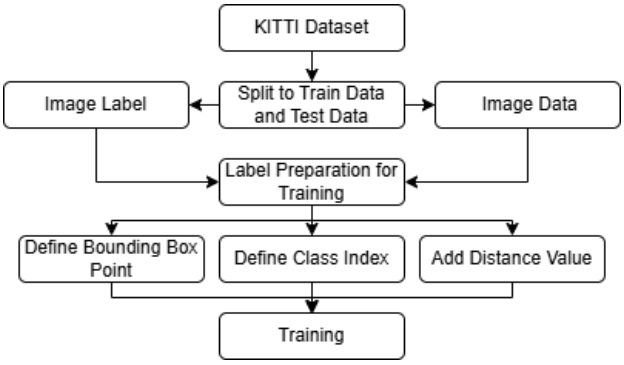


Figure 6: Pre-processing data

The employment of an additional layer in the form of an attention mechanism has been proven to reduce the loss value to a greater extent than without adding this layer. Figure 7 designates the difference in loss values with the results of train loss and validation loss reaching 16,031 and 17,565. These results are quite different from the pre-trained model results of Vajgl et al. [10]. The next comparison is related to the Pascal VOC AP (Average Precision) evaluation measurement for each class. These measurements are usually applied to evaluate the performance of object detection models [33]. This is an attempt to enhance the proposed SACPS detecting model.

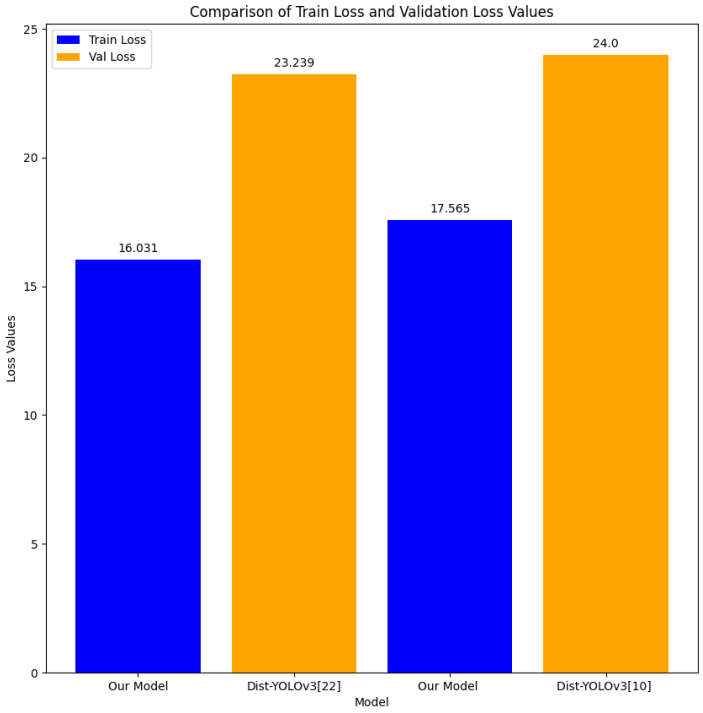


Figure 7: Comparison of train loss and validation loss

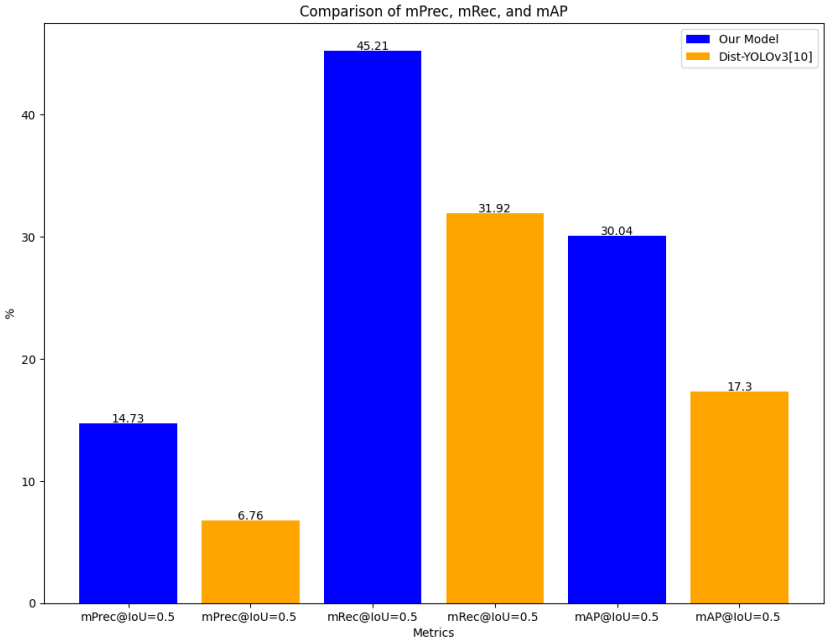


Figure 8: Comparison of mPrec, mRec, and mAP values

Figure 8 indicates a comparative detecting performance based on Pascal VOC AP measurements with a threshold or Intersection over Union (IoU) of 0.5. The model developed in this study was superior in evaluation results with mPrec, mRec, and mAP values of 14.73%, 45.21%, and 30.04% respectively observed. This proves that the pre-trained model can detect objects that exist properly. Hence, it enables to increase in the mAP value greater than the original model [10] available on public links. However, due to the limitations of existing devices, we can only use a batch number of 2 so the training process is time-consuming.

To further evaluate the optimization behavior of the proposed model, we conducted a convergence analysis based on the training and validation loss values recorded over 40 epochs. This analysis aims to assess whether the training process led to stable convergence and to identify any signs of overfitting or instability that might affect the model's generalization capability.

As depicted in the loss convergence curves in Figure 9, the training loss decreased steadily and stabilized at approximately 16.03, indicating proper convergence. Although the validation loss initially started at a high value due to uncalibrated predictions, it consistently declined over 40 epochs, eventually reaching a value of 23.24. This trend confirms the model's effective learning of generalizable features without significant overfitting. The relatively high absolute loss values are attributed to the multi-objective nature of the loss function and unnormalized scaling.

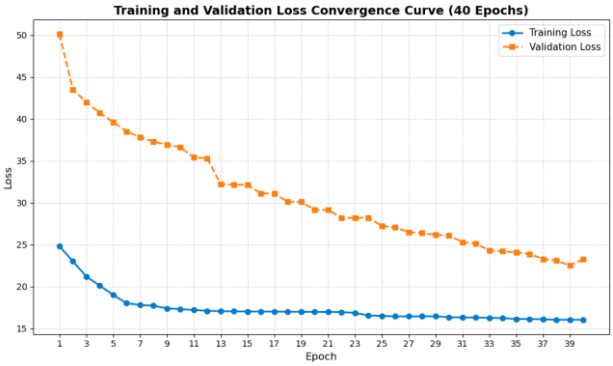


Figure 9: Training and validation loss convergence curve

After reaching a better pre-trained model compared to the pre-trained model [10], we then carried out experiments on context knowledge and adaptive requirements on $C_1, C_3 \in \{C_{V3}, C_{V1}\}$. The experiment was conducted by detecting images with low light intensity. The utilized light intensity level was based on the average RGB value. Hence, we gained an equation to calculate the increased new light intensity in equation (5).

$$I_{new} = I_{old} \times F \tag{5}$$

Where I_{new} was the new intensity updated based on I_{old} multiplied by F, where F was the factor enhancing the light intensity with F > 1, equation (5) was employed if the $I_{old} < 50$. Also, it manifested which value was taken from the average light intensity value in the images. Furthermore, the F value would be adjusted to the old intensity value detected with $F \in \{F > 1\}$. Briefly, we conducted experiments with images indicating an average light intensity below 50. Technically, Table 6 describes the changing results of the intensity of the utilized images.

The experiment in Table 6 discloses good adaptive results because the improvement in the intensity value was adjusted to the previous light intensity value. This was aimed at avoiding the damage of the obtained information results in the detected images. In addition, excessively increasing the intensity could damage the existing color composition and could create noise in the image. The scenario of $C_1, C_3 \in \{C_{V3}, C_{V1}\}$ was effectively performed to increase the light intensity in the image on average by 100.0703.

Figure 10 shows increased light intensity conducted by the system based on miscellaneous schemes from C_3 containing various values of old brightness (I_{old}). The resulting new brightness (I_{new}) would not be increased excessively even though the I_{old} value was close to the value of 50 as represented by Table 6 in the 25th sample. The I_new value produced a fairly stable value if $I_{old} \in \{I_{old} > 25\}$. However, the I_{new} value underwent unstable changes when the $I_{old} \in \{I_{old} < 10\}$ value. The stability of the I_newvalue has a very prominent role in the final results of object and distance predictions. This was proven in the 4th sample experiment for $I_{old} < 10$ and the 17th sample for $I_{old} > 25$.

Table 6:
Results of adaptation of C_{V3} light settings

No. Sample	Light Intensity before (I_{old})	Light Intensity after (I_{new})
1	3.3443	67.4892
2	5.5522	83.6164
3	7.7493	77.9420
4	8.7250	87.4341
5	9.9340	99.1287
6	11.1267	67.5903
7	14.2915	86.0231
8	16.4727	98.8303
9	18.6484	110.8143
10	19.8362	113.6794
11	22.0120	88.2736
12	25.3866	101.8137
13	27.5531	109.3286
14	28.5278	111.4813
15	29.7263	113.5586
16	31.8851	95.8494
17	36.2576	108.2096
18	37.4274	110.3538
19	38.4252	112.0422
20	39.6189	113.5254
21	41.8004	100.2741
22	45.1630	107.8529
23	47.3357	110.9795
24	48.3229	112.2473
25	49.5122	113.4192
Average light intensity after (I_{new})		100.0703

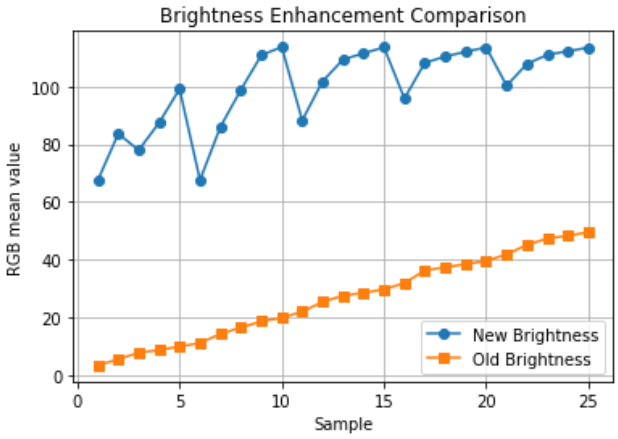


Figure 10: Brightness enhancement comparison

(a)

(b)

Figure 11: (a) 4th sample with a light intensity value of 8.7250. (b) 17th sample with a light intensity value of 36.2576.

Figure 11 illustrates the input images of the 4th and 17th samples, both of which have undergone light intensity enhancement. These two examples represent different segments of the I_{old} range: the 4th sample with $I_{old} < 10$, and the 17th sample with $I_{old} \approx 45$. As depicted in Figure 13, the subsequent outputs—object prediction and distance estimation—exhibited markedly different behaviors under the same enhancement procedure. In the case of the 4th sample, the original image was extremely dark, making essential visual features such as edges and textures virtually indiscernible. Upon applying light intensity scaling, the image became noticeably brighter, but at the cost of amplifying noise and artifacts. This led to unstable or erroneous object detection, with the model often missing or misclassifying key targets.

Furthermore, distance prediction in this sample was unreliable due to the distortion introduced by aggressive enhancement from such a low baseline. By contrast, the 17th sample, with moderately low lighting, retained a sufficient amount of visual information. The light adjustment process yielded a stable and visually coherent image (I_{new}), resulting in more accurate object detection and consistent distance estimation. This comparison demonstrates that while the proposed model exhibits

improved robustness under typical low-light conditions, its performance significantly deteriorates when the initial image intensity falls below a critical threshold (approximately I_{old} < 10). Although the implemented brightness enhancement method—using a scaling factor FFF when I_{old} < 50—effectively improves visibility in low-light scenarios, it remains a rule-based and non-learning approach. This mechanism was chosen to demonstrate a lightweight, self-triggered adaptation behavior within the SACPS context. However, the instability observed in extremely dark cases highlights the limitations of fixed-rule strategies. Future improvements will involve integrating adaptive, learning-based enhancement modules (e.g., LLNet, Zero-DCE, or transformer-based models) that provide more context-aware correction while mitigating noise and preserving structural detail, thereby enhancing both object recognition and distance estimation reliability in diverse lighting conditions.

To demonstrate that the proposed model outperforms its predecessors, we conducted an ablation study on four model configurations: Dist-YOLOv3 [10], Dist-YOLOv3+Xception, Dist-YOLOv3+Attention, and Dist-YOLOv3+Xception+Attention (Proposed Method). Furthermore, to support the model's generalisation capability, we evaluated its performance under varying lighting conditions by simulating reduced illumination on the KITTI test data. The results of this ablation study are presented in Table 7.

Table 7:
Ablation study

Model Variant	Light Intensity	mAP (%)	mPrec (%)	mRec (%)
Dist-YOLOv3 [10]	100%	17.30	6.76	31.92
	80%	15.48	5.97	29.04
	60%	13.12	4.88	26.04
	40%	10.93	3.75	22.01
Dist-YOLOv3 + Xception	100%	23.61	11.25	37.88
	80%	21.30	9.94	34.41
	60%	18.42	8.11	30.75
	40%	15.67	6.42	26.08
Dist-YOLOv3 + Attention	100%	22.63	10.63	35.82
	80%	19.80	8.90	31.99
	60%	17.01	7.34	27.45
	40%	14.83	6.09	25.17
Dist-YOLOv3 + Xception + Attention	100%	30.04	14.73	45.21
	80%	27.46	13.06	41.00
	60%	24.85	11.30	36.48
	40%	21.78	9.64	31.59

The results indicate that both Xception and the attention mechanism independently improve performance over the baseline under all lighting conditions. The Xception architecture in Dist-YOLOv3 proves especially beneficial in low-light scenarios. The complete model, which combines both Xception and attention, achieves the best overall results, demonstrating the effectiveness and robustness of the proposed approach in adaptive object recognition tasks.

To fulfill the context knowledge and adaptive requirements in $C_1, C_2, C_3 \vee C_1, C_2 \in \{C_{V1}, C_{V2}, C_{V3}, C_{V4}\}$, the system should be able to produce a sound output of notification in the form of identified objects in front of it. These requirements were built through the Google Text to Speech (gTTS) library covering a sample rate of 22050-44100 Hz. As a result, it provides clear sound with clear quality of each pronounced word. A notification will come out of the system if there is an object that is 5 meters away in front of you. Figure 12 demonstrates the sound waves resulting from the detection of objects located 5 meters in front of the machine.

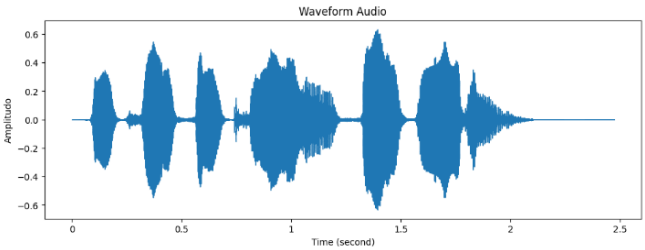


Figure 12: Audio waveforms from system output

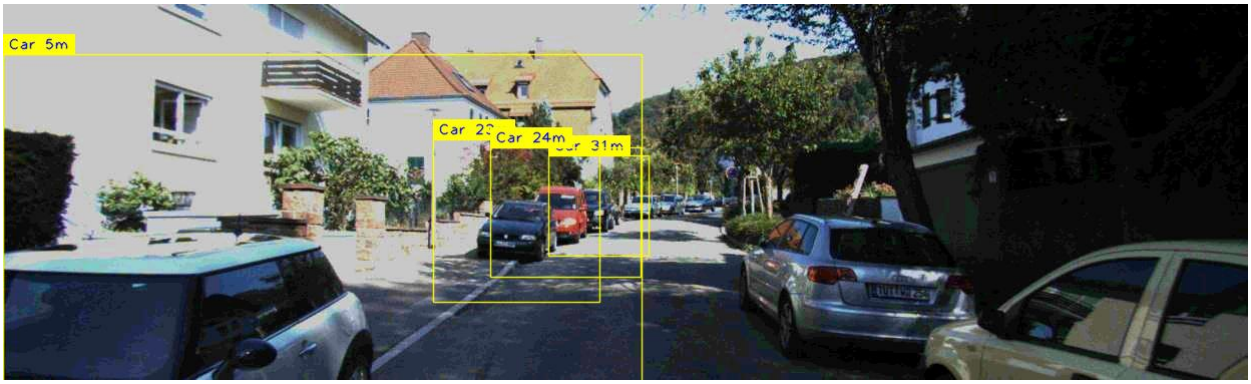
The audio waves in Figure 12 illustrate that there is a car object 5 meters in front of the camera. This indicates one of the successful final results of detecting and fulfilling the needs of $C_1, C_2, C_3 \vee C_1, C_2 \in \{C_{V1}, C_{V2}, C_{V3}, C_{V4}\}$ by all the experimental scenarios cultivated in this study.

Internal validity refers to the extent to which research results can be attributed to the manipulation of independent variables rather than other factors. The threat to internal validity in this study refers to the irrelevant use of data in various environments. In this case, these data refer to the adopted data for the object recognition model training process. For this reason, it may cause significant accuracy gaps and reduce the adaptability of the system in diverse environments.

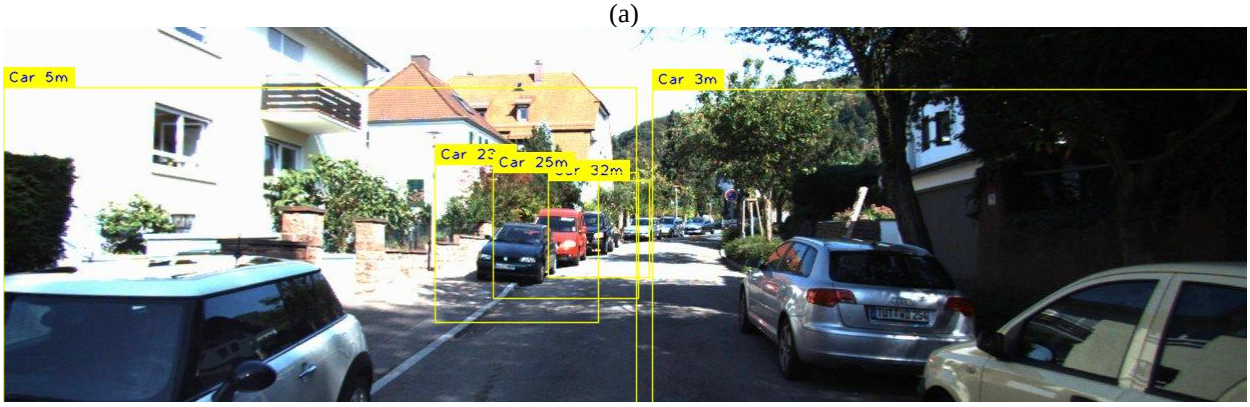
Construct validity refers to the extent to which a test or measuring instrument measures a particular concept or construct. Threats to construct validity encompass inadequate definitions and supportive effects. The threat of inadequate definitions is caused by very limited object classes and the adaptation of the existing classes in the KITTI dataset. As a result, the model is unable to recognize objects comprehensively. Besides, this will also cause the inaccurate object recognition model's metric measurements. One threat to the supportive effect is that if other factors (e.g. voice assistance or user interaction with additional devices) assist in object recognition, the results may not fully reflect the capabilities of the object recognition model itself.

Construct validity refers to the extent to which a test or measuring instrument measures a particular concept or construct. Threats to construct validity encompass inadequate definitions and supportive effects. The threat of inadequate definitions is caused by very limited object classes and the adaptation of the existing classes in the

KITTI dataset. As a result, the model is unable to recognize objects comprehensively. Besides, this will also cause the inaccurate object recognition model's metric measurements. One threat to the supportive effect is that if other factors (e.g. voice assistance or user interaction with additional devices) assist in object recognition, the results may not fully reflect the capabilities of the object recognition model itself.



(a)



(b)

Figure 13: (a) Results of increased light intensity, object prediction, and distance prediction in the 4th sample. (b) Results of increasing light intensity, object prediction, and distance prediction on the 17th sample

The evaluative results, in terms of object recognition and adaptation, yielded excellent results when measured by the corresponding metrics. Conversely, this has not ruled out the possibility of threats to the conclusive validity. Although it can be inferred that the object recognition model produced by this research exhibits better performance than previous models, several cases highlight the model's shortcomings. One of them is when objects are closely situated to each other or partially captured by the camera. This makes the model unable to accurately recognize what object is being detected. Furthermore, if adaptive ability is considered successful, there is a possibility of adaptive failures when the light intensity is increased. When the adaptation process for increasing light intensity occurs, the results occasionally indicate an effect on the image, such as loss of quality in miscellaneous aspects.

The limited number of employed object classes and the existing dataset environment have opened new threats to external validity. Even though the entire evaluation results of this research are auspicious, it needs to be re-emphasized that testing miscellaneous environmental characteristics depends on the data used, so it is highly recommended to make improvements by adding to the data used, either by adding data from independent collection results or by combining it with the available dataset.

5 Discussions

Our model introduces two main architectural enhancements to the original Dist-YOLOv3: replacing the Darknet53 backbone with the Xception architecture and integrating an attention mechanism layer before the head. These improvements have shown a notable increase in detection performance. In terms of quantitative evaluation, our model achieves a mean Average Precision (mAP) of 30.04%, compared to the baseline mAP of approximately 17.3% reported by Vajgl et al. [10], marking a 12.74% improvement. Additionally, the mean recall increased to 45.21%, and the mean precision reached 14.73%. We compared the pre-trained model from this study with the pre-trained model available at https://gitlab.com/EnginCZ/yolo-with-distance. These metrics demonstrate a better detection capability in real-world scenarios, especially in complex environments with occlusions or variable lighting conditions. The improvement is primarily due to:

- Xception Backbone: With its depthwise separable convolutions, Xception enables deeper and more efficient feature extraction than the original Darknet53. This enables the model to capture fine-grained spatial features more accurately.

- Attention Layer: The attention mechanism selectively emphasizes relevant spatial features and suppresses irrelevant or noisy ones, improving both classification and localization accuracy.

However, the performance gain comes at a cost. The model incurs higher computational complexity, especially during training, and exhibits higher loss values (16.031 training loss and 17.565 validation loss). This is attributed to:

- The use of a small batch size (2) due to limited hardware resources, which affects training stability and convergence speed. The inclusion of a distance loss term, which adds to the total loss function and increases its numerical magnitude.

- The use of historical label-based distance data from KITTI may not perfectly align with real-time physical constraints.

In addition to detection accuracy, our model also incorporates a self-adaptive light intensity enhancement mechanism to address challenges in low-light environments. This component is crucial for ensuring visual clarity and maintaining object recognition performance in varying illumination conditions. The experimental results indicate that the brightness enhancement system successfully increases image intensity in dark environments, with an average post-adaptation brightness of 100.07, up from initial values of less than 50. This enhancement is designed to avoid overexposure by applying a proportional intensity factor based on the original image brightness. Two representative cases were highlighted:

- In the 4th sample (initial brightness: 8.72), although the enhancement increased visibility, it also introduced noticeable noise and partial information loss, which negatively affected detection accuracy.

- In the 17th sample (initial brightness: 36.26), the enhancement achieved more stable brightness with minimal noise, leading to more accurate object and distance predictions.

These findings reveal a critical insight: while adaptive brightness enhancement is beneficial, it must be applied carefully depending on the original light level. Excessive enhancement in extremely dark images may degrade the image quality and impair the detection process. This aspect reflects a trade-off between adaptation robustness and prediction accuracy. In future iterations, incorporating learned enhancement filters or adaptive gain control mechanisms may help optimize this process and reduce noise in low-light scenarios.

Despite the observed improvements, several limitations remain in the current implementation. First, tuning hyperparameters—particularly those related to attention mechanisms and light adaptation thresholds—required a series of empirical iterations due to the absence of prior benchmarks in this domain. This process introduces complexity and may compromise reproducibility if thorough documentation is not provided. Second, although the attention layer improves detection accuracy, it inevitably increases computational load during inference, which could impact responsiveness—especially on low-power wearable platforms. While inference time was not formally measured in this study, the additional layers are expected to introduce some delay due to their sequential nature and computational requirements. Lastly, deployment in varied environments presents practical challenges: indoor scenes often suffer from occlusion and low-light noise, while outdoor settings may include background clutter and high dynamic range lighting. These differences complicate generalization and suggest the need for environment-aware calibration in future iterations.

6 Conclusion and further studies

This paper introduces a CPS-based object recognition model with automatic adaptive capabilities to address uncertain real-world objects in a CPS environment characterized by adaptive strategies. The main focus of the experiment is on the adaptive strategies of the CPS-based object recognition system and the performing specifications of the system. The research results report that this system can make object recognition better, notably for blind people during navigating practices. This experiment was carried out in the context of application development in the field of self-adaptive cyber-physical systems (SACPS), particularly in smart glasses. The utilization of adaptive strategies (e.g. increasing light intensity and object recognition based on context) has been fruitful. Increased light intensity is carefully processed to maintain image quality and avoid noise.

Moreover, this study has successfully developed a machine learning/deep learning-based object recognition model superior to previous models. This model can amplify the mAP value and accuracy value in recognizing objects. Even though there are limitations to the applied devices, this improvement signifies a highly significant result. In particular, the experimental results also denote that the system can issue a sound notification to its users when there is an object within five meters in front of them. Given this fact, it meets the adaptive requirements in a CPS environment. Overall, this study has fruitfully developed an adaptive CPS-based object recognition model that can advance the quality of object recognition, notably for blind people in the context of smart glasses. The results of this scrutiny showcase remarkable

potentials in cultivating the life quality of the blind people while performing daily activities.

Based on the entire experimental results, some major findings attracted our attention. As an example, at the detecting evaluative model stage, the comparison of train loss and validation loss revealed better results than the detecting model of Vajgl et al. [10] with average reduction values of 26.7% and 26.8% respectively. Besides, the Pascal VOC AP metric indicated that the values of each metric component had increased for both the mPrec, mRec, and mAP elements with an average increase of 7.97%, 13.29%, and 12.74%, respectively. Moreover, the adaptability of our model was represented in the form of adaptation to changes in light intensity. The evaluative results indicated that, after the adaptation process, the average light intensity of the enhanced images reached 100.0703 (I_{new}). This reflects the ability of the proposed adaptive mechanism to consistently raise low-light image brightness to an optimal level. Such enhancement significantly contributed to the improved functionality and adaptability of the object detection model, particularly in varying illumination conditions.

Our proposed adaptive strategy has an indispensable role in the real-world object recognition process based on broader cues in multiple quality attributes. Among them, precision, recall, and average accuracy are determined based on the ability to adapt to the surrounding environment. This indicates that efforts to resolve the problems of recognizing real-world objects have been restricted to certain cues without adaptability.

This paper opens up new opportunities for future investigation. To illustrate, the adaptive strategy of a CPS-based object recognition system can be directly employed based on an effective adaptive environment. In this experiment, we applied the model to the needs of a smart glasses system. However, it does not rule out the possibility of being applied to other system needs in the real world. Our model has the potential to be applied to several applications, including autonomous driving, robot vision, smart doors, and so on.

Shortly, we plan to conduct further experiments to optimize this adaptive strategy of object recognition to invigorate its accuracy and performance through a lighter model when being implemented. As an illustration, improving adaptability through the addition of adaptive features (e.g. system failure management, self-scheduled recognition model updates, and identification of new object classes) to increase system knowledge. In addition, replacing the YOLO-based model with the latest version is one way to optimize the object recognition model. Applying automatic parameter selection methods in the training process is the primary step in enhancing such an object recognition model. Further, we will also upgrade the designed smart glasses device to be able to provide thorough and specific features for other needs of blind people. As part of this plan, we aim to conduct a pilot usability study involving individuals who are blind or visually impaired, in collaboration with assistive technology communities. This will allow us to evaluate the system's real-world applicability, user satisfaction, and interaction flow, which are essential for ensuring its practicality and accessibility. In addition, we will validate the system in more realistic environments for visually impaired users by employing data from indoor navigation or close-range object interaction, which are more representative of real-world scenarios faced during daily activities.

The impacts and importance of our proposed model are related to the sustainability of object recognition systems for long-term needs. Considering that the variety of real-world cues recognized by a recognition system has the potential to undergo rapid and unpredictable changes. Our model offers adaptability to adjust the state of the system based on specific contexts, notably increased accuracy in recognizing objects affected by light intensity. Also, this represents an achievement of the objectives of this work, namely developing a generic model for adapting CPS services when recognizing real-world objects as heterogeneous and strongly influenced by their environmental conditions.

References

[1] Hikmawati, N. U. Maulidevi, and K. Surendro, "Adaptive rule: A novel framework for recommender system," *ICT Express*, vol. 6, no. 3, pp. 214–219, 2020, doi: 10.1016/j.icte.2020.06.001.

[2] P. O. Antonino, A. Morgenstern, B. Kallweit, M. Becker, and T. Kuhn, "Straightforward Specification of Adaptation-Architecture-Significant Requirements of IoT-enabled Cyber-Physical Systems," *Proc. - 2018 IEEE 15th Int. Conf. Softw. Archit. Companion, ICSA-C 2018*, pp. 19–26, 2018, doi: 10.1109/ICSA-C.2018.00012.

[3] S. Zeadally, T. Sanislav, and G. D. Mois, "Self-Adaptation Techniques in Cyber-Physical Systems (CPSs)," *IEEE Access*, vol. 7, pp. 171126–171139, 2019, doi: 10.1109/ACCESS.2019.2956124.

[4] A. Petrovska and A. Pretschner, "Learning Approach for Smart Self-Adaptive Cyber-Physical Systems," in *2019 IEEE 4th International Workshops on Foundations and Applications of Self* Systems (FAS*W)*, 2019, pp. 234–236. doi: 10.1109/FAS-W.2019.00061.

[5] E. Zavala, X. Franch, J. Marco, A. Knauss, and D. Damian, "SACRE: Supporting contextual requirements' adaptation in modern self-adaptive systems in the presence of uncertainty at runtime," *Expert Syst. Appl.*, vol. 98, pp. 166–188, 2018, doi: 10.1016/j.eswa.2018.01.009.

[6] A. Petrovska, S. Quijano, I. Gerostathopoulos, and A. Pretschner, "Knowledge aggregation with subjective logic in multi-agent self-adaptive cyber-physical systems," *Proc. - 2020 IEEE/ACM 15th Int. Symp. Softw. Eng. Adapt. Self-Managing Syst. SEAMS 2020*, no. June, pp. 149–155, 2020, doi: 10.1145/3387939.3391600.

[7] D. Gürdür and F. Asplund, "A systematic review to merge discourses: Interoperability, integration and cyber-physical systems," *J. Ind. Inf. Integr.*, vol. 9, no. October, pp. 14–23, 2018, doi:

10.1016/j.jii.2017.12.001.

[8] Y. Eslami, M. Lezoche, P. Kalitine, and S. Ashouri, "How the cooperative cyber physical enterprise information systems (CCPEIS) improve the semantic interoperability in the domain of industry 4.0 through the knowledge formalization," *IFAC-PapersOnLine*, vol. 54, no. 1, pp. 924–929, 2021, doi: 10.1016/j.ifacol.2021.08.110.

[9] H. Nakagawa, A. Ohsuga, and S. Honiden, "Towards dynamic evolution of self-Adaptive systems based on dynamic updating of control loops," *Int. Conf. Self-Adaptive Self-Organizing Syst. SASO*, pp. 59–68, 2012, doi: 10.1109/SASO.2012.17.

[10] M. Vajgl, P. Hurtik, and T. Nejezchleba, "Dist-YOLO: Fast Object Detection with Distance Estimation," *Appl. Sci.*, vol. 12, no. 3, pp. 1–13, 2022, doi: 10.3390/app12031354.

[11] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 1800–1807, 2017, doi: 10.1109/CVPR.2017.195.

[12] H. Walle, C. De Runz, B. Serres, and G. Venturini, "A Survey on Recent Advances in AI and Vision-Based Methods for Helping and Guiding Visually Impaired People," *Appl. Sci.*, vol. 12, no. 5, 2022, doi: 10.3390/app12052308.

[13] J. R. Jiang, "An improved cyber-physical systems architecture for Industry 4.0 smart factories," *Adv. Mech. Eng.*, vol. 10, no. 6, pp. 918–920, 2018, doi: 10.1177/1687814018784192.

[14] M. K. Habib and C. Chimsom I, "CPS: Role, Characteristics, Architectures and Future Potentials," *Procedia Comput. Sci.*, vol. 200, pp. 1347–1358, 2022, doi: 10.1016/j.procs.2022.01.336.

[15] J. Zhou, Y. Huang, and B. Yu, "Mapping Vegetation-Covered Urban Surfaces Using Seeded Region Growing in Visible-NIR Air Photos," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 8, no. 5, pp. 2212–2221, 2015, doi: 10.1109/JSTARS.2014.2362308.

[16] M. Búr, G. Szilágyi, A. Vörös, and D. Varró, "Distributed graph queries over models@run.time for runtime monitoring of cyber-physical systems," *Int. J. Softw. Tools Technol. Transf.*, vol. 22, no. 1, pp. 79–102, 2020, doi: 10.1007/s10009-019-00531-5.

[17] D. Weyns *et al.*, "Towards Better Adaptive Systems by Combining MAPE, Control Theory, and Machine Learning," *Proc. - 2021 Int. Symp. Softw. Eng. Adapt. Self-Managing Syst. SEAMS 2021*, no. 1, pp. 217–223, 2021, doi: 10.1109/SEAMS51251.2021.00036.

[18] S. H. Ahmed, G. Kim, and D. Kim, "Cyber Physical System: Architecture, applications and research challenges," *IFIP Wirel. Days*, no. November, 2013, doi: 10.1109/WD.2013.6686528.

[19] M. Sony, "Design of cyber physical system architecture for industry 4.0 through lean six sigma: conceptual foundations and research issues," *Prod. Manuf. Res.*, vol. 8, no. 1, pp. 158–181, 2020, doi: 10.1080/21693277.2020.1774814.

[20] T. Kluge, "A role-based architecture for self-adaptive cyber-physical systems," *Proc. - 2020 IEEE/ACM 15th Int. Symp. Softw. Eng. Adapt. Self-Managing Syst. SEAMS 2020*, pp. 120–124, 2020, doi: 10.1145/3387939.3391601.

[21] G. Weichhart, J. Mangler, C. Mayr-Dorn, A. Egyed, and A. Hämmerle, "Production process interoperability for cyber-physical production systems," *IFAC-PapersOnLine*, vol. 54, no. 1, pp. 906–911, 2021, doi: 10.1016/j.ifacol.2021.08.188.

[22] R. Casadei, A. Placuzzi, M. Viroli, and D. Weyns, "Augmented Collective Digital Twins for Self-Organising Cyber-Physical Systems," *Proc. - 2021 IEEE Int. Conf. Auton. Comput. Self-Organizing Syst. Companion, ACSOS-C 2021*, pp. 160–165, 2021, doi: 10.1109/ACSOS-C52956.2021.00051.

[23] A. Aradea, R. Rianto, and H. Mubarok, "Inference Model for Self-Adaptive IoT Service Systems," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 4, pp. 337–349, 2021, doi: 10.22266/ijies2021.0831.30.

[24] Aradea, I. Supriana, and K. Surendro, "ARAS: adaptation requirements for adaptive systems," *Autom. Softw. Eng.*, vol. 30, no. 1, p. 2, 2022, doi: 10.1007/s10515-022-00369-3.

[25] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018, [Online]. Available: http://arxiv.org/abs/1804.02767

[26] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[27] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. doi: 10.1007/978-3-642-29044-2.

[28] M. M. John, H. H. Olsson, and J. Bosch, "Developing ML/DL Models: A Design Framework," in *2020 IEEE/ACM International Conference on Software and System Processes (ICSSP)*, 2020, pp. 1–10.

[29] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361. doi: 10.1109/CVPR.2012.6248074.

[30] K. Lu, F. Zhao, X. Xu, and Y. Zhang, "An object detection algorithm combining self-attention and YOLOv4 in traffic scene," *PLoS One*, vol. 18, no. 5 May, pp. 1–18, 2023, doi: 10.1371/journal.pone.0285654.

[31] L. Chen, W. Shi, and D. Deng, "Improved yolov3 based on attention mechanism for fast and accurate ship detection in optical remote sensing images," *Remote Sens.*, vol. 13, no. 4, pp. 1–18, 2021, doi: 10.3390/rs13040660.

[32] J. Sun, H. Ge, and Z. Zhang, "AS-YOLO: An Improved YOLOv4 based on Attention Mechanism and SqueezeNet for Person Detection," *2021 IEEE 5th Adv. Inf. Technol. Electron. Autom. Control*

Conf., vol. 5, pp. 1451–1456, 2021, [Online]. Available: https://api.semanticscholar.org/CorpusID:233198337.

[33] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 and Beyond," pp. 1–34, 2023, [Online]. Available: http://arxiv.org/abs/2304.00501.

[34] B. Dhanalakshmi and P. Tamije Selvy, "Enhancing Predictive Capabilities for Cyber Physical Systems Through Supervised Learning," Inform., vol. 49, no. 16, pp. 77–86, 2025, doi: 10.31449/inf.v49i16.7635.

[35] N. Azeri, O. Hioual, and O. Hioual, "Efficient Vanilla Split Learning for Privacy-Preserving Collaboration in Resource-Constrained Cyber-Physical Systems," Inform., vol. 48, no. 11, pp. 167–180, 2024, doi: 10.31449/inf.v48i11.6186.