applied sciences

MDPI

Article

A Staged Framework for Computer Vision Education: Integrating AI, Data Science, and Computational Thinking

In-Seong Jeon [1], Sukjae Joshua Kang [2] and Seong-Joo Kang [3,*]

[1] Department of Computer Education, Korea National University of Education, Cheongju 28173, Republic of Korea; jinsung4069@knue.ac.kr
[2] Peptide Biology Laboratory, The Salk Institute for Biological Studies, La Jolla, CA 92037, USA; sjkang@salk.edu
[3] Department of Chemistry Education, Korea National University of Education, Cheongju 28173, Republic of Korea
* Correspondence: sjkang@knue.ac.kr

Abstract: Computer vision education is increasingly important in modern technology curricula; yet, it often lacks a systematic approach integrating both theoretical concepts and practical applications. This study proposes a staged framework for computer vision education designed to progressively build learners' competencies across four levels. This study proposes a four-staged framework for computer vision education, progressively introducing concepts from basic image recognition to advanced video analysis. Validity assessments were conducted twice with 25 experts in the field of AI education and curricula. The results indicated high validity of the staged framework. Additionally, a pilot program, applying computer vision to acid–base titration activities, was implemented with 40 upper secondary school students to evaluate the effectiveness of the staged framework. The pilot program showed significant improvements in students' understanding and interest in both computer vision and scientific inquiry. This research contributes to the AI educational field by offering a structured, adaptable approach to computer vision education, integrating AI, data science, and computational thinking. It provides educators with a structured guide for implementing progressive, hands-on learning experiences in computer vision, while also highlighting areas for future research and improvement in educational methodologies.

Keywords: computer vision education; staged framework; computational thinking; data science; symbolic AI; neural network-based AI

check for updates

Citation: Jeon, I.-S.; Kang, S.J.; Kang, S.-J. A Staged Framework for Computer Vision Education: Integrating AI, Data Science, and Computational Thinking. *Appl. Sci.* **2024**, *14*, 9792. https://doi.org/10.3390/app14219792

Academic Editors: Bikram Bhuyan, Manolo Dulva Hina and Amar Ramdane-Cherif

Received: 26 September 2024
Revised: 19 October 2024
Accepted: 24 October 2024
Published: 26 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Over the past few years, advancements in computer vision technology have accelerated, largely due to deep learning developments. Computer vision has emerged as an important branch of artificial intelligence, with diverse applications in image recognition, object detection, and video analysis. Examples of these applications include analyzing emotions through facial expression detection [1], student behavior monitoring systems [2], and computer vision tasks using deep active learning [3]. The growth of neural network-based AI, particularly deep learning, has driven innovation in computer vision, leading to its widespread adoption in industries such as medical image analysis, autonomous driving, and security systems. Deep learning algorithms, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Generative Adversarial Networks (GANs), have shown remarkable performance in image classification, object detection, and image generation.

Neural network-based AI is adept at addressing complex problems by learning patterns from data, it often lacks transparency and logical reasoning capabilities. On the other hand, symbolic AI, sometimes called rule-based or classical AI, works with high-level symbolic representations using clear rules and logic for data processing [4]. Symbolic AI is

particularly effective for tasks involving logical reasoning, knowledge representation, and the manipulation of abstract ideas, which makes it suitable for areas such as expert systems and language comprehension [5]. However, symbolic AI faces challenges in learning from large amounts of raw data and adapting to new, unseen situations due to its reliance on predefined rules and structures [6].

Computer vision technology is critical to both present and future information technology industries, and its study is fundamental for students. In an environment where neural network-based AI, such as deep learning, is central to computer vision technology, students need to develop practical skills to solve problems using real data, in addition to theoretical knowledge. However, most current educational curricula are either theory-centric or overly focused on practice, lacking a systematic educational approach that integrates neural network-based AI and symbolic AI. Despite technological advancements, there is a lack of systematic educational frameworks capable of effectively teaching computer vision technologies [3]. Especially in primary and secondary education, teaching computer vision requires a staged learning approach that aligns with students' levels and educational goals.

The integration of data science and computational thinking is essential in computer vision education. Data science provides the foundation for collecting, preprocessing, analyzing, and modeling image data, enabling students to learn and evaluate computer vision models [7]. Computational thinking plays a vital role in developing students' abilities to systematically solve problems. Elements of computational thinking such as decomposition, abstraction, pattern recognition, and algorithm design help students logically approach computer vision problems [8].

This study aims to develop a staged framework for computer vision education by integrating neural network-based AI and symbolic AI in each stage. This framework incorporates AI, data science, and computational thinking to enhance problem-solving abilities. By emphasizing a staged framework, we design each stage to allow learners to simultaneously experience theory and practice through this integration.

2. Theoretical Background

2.1. Artificial Intelligence, Data Science, and Education

The rapid advancement of artificial intelligence (AI) has revolutionized various industries, necessitating its integration into educational curricula. Particularly, neural network-based AI, such as deep learning, has shown exceptional performance in learning patterns from large-scale data and solving complex problems [9]. However, these technologies often lack explainability and logical reasoning abilities, making them somewhat abstract and difficult for learners to comprehend from an educational perspective. In contrast, symbolic AI excels in knowledge representation and logical reasoning, allowing for clear explanations and interpretations [4].

Data science is instrumental in understanding and utilizing these AI technologies. It encompasses processes such as data collection, preprocessing, analysis, visualization, modeling, and evaluation, providing students with foundational and essential skills needed to handle real-world data. In fields such as computer vision, data science provides the knowledge base required for effective image data processing. Incorporating data science into the curriculum enables students to apply computer vision techniques to real-world issues [7]. For instance, Ref. [10] points out the importance of well-designed educational datasets tailored to different school levels in AI and data science education, highlighting how step-by-step experiences in data science processes can enhance learning outcomes. Likewise, Ref. [11] found that integrating data science and machine learning into scientific inquiry enables students to apply AI technologies to domain-specific problems, fostering practical problem-solving skills.

The data science process involves six main stages, ranging from data collection to model deployment. Each stage is essential in helping students acquire the skills necessary to manage and analyze data effectively, providing a basis for their understanding and use of AI technologies, as shown in Table 1.

Table 1. Stages of data science process.

Stage	Element	Description and Related Research
Stage 1	Data Collection	• Gathering necessary data for a project from various sources, including structured data (e.g., databases) and unstructured data (e.g., text, images). • The importance of data collection is emphasized in many studies, as data quality significantly impacts analysis results [12].
Stage 2	Data Preprocessing	• Transforming collected data into a suitable format for analysis, including data cleaning, handling missing values, and data transformation. • Data preprocessing is essential to ensure data accuracy and consistency, highlighted in various research [13].
Stage 3	Data Exploration	• Visualizing and analyzing data to discover meaningful patterns and insights. It is crucial for understanding the characteristics of the data. • Data exploration is foundational in data analysis for understanding data structure and distribution [14].
Stage 4	Modeling	• Building predictive models based on data using machine learning algorithms such as regression, classification, and clustering. • Modeling is the core of data science projects, where predictive models are constructed using various machine learning techniques [15].
Stage 5	Model Evaluation	• Assessing the performance of the built model and validating its accuracy. It is important for confirming and improving the model's validity. • Model evaluation is critical for measuring model performance and identifying areas for improvement [16].
Stage 6	Model Deployment	• Implementing the final model in a real-world environment to perform real-time predictions. It is the last step to make the model usable in practice. • Model deployment prepares the model for use in practical settings, emphasized in various studies [17].

2.2. Computer Vision and Education

Computer vision is a subfield of artificial intelligence that enables computers to interpret and understand visual data. It encompasses several stages, including image recognition, object detection, image segmentation, and video analysis, each employing various algorithms and neural network models. Convolutional Neural Networks (CNNs) are the most widely used deep learning models in computer vision, demonstrating excellent performance in extracting features from images and classifying objects [9].

Image recognition is the foundational stage of computer vision, where computers identify and classify specific objects or patterns within images. CNNs are primarily used for this purpose, capable of accurately classifying and recognizing objects based on large-scale training data. Object detection involves identifying and locating specific objects within an image, with algorithms like You Only Look Once (YOLO) being representative models known for efficient object detection [18].

Image segmentation is the process of partitioning an image into meaningful regions, distinguishing object boundaries or separating the foreground from the background. This technology is utilized in various fields, such as medical image analysis and autonomous driving, with models like U-Net being commonly employed [19]. Lastly, video analysis involves tracking objects and recognizing actions in sequential image frames, analyzing temporal changes and object movements [20]. These stages of computer vision, along with their key elements and related research, are summarized in Table 2. These computer vision technologies have become essential learning topics, enabling students to develop the ability to understand and interpret visual data.

Computer vision education is a key aspect of AI education and can offer more effective learning by integrating data science and coding education. While computer vision is inherently complex, developing an educational framework that systematically teaches these technologies allows students to understand the essence of the technology and cultivate practical application skills. This study proposes a staged approach to computer vision

education, aiming to help learners progressively acquire knowledge from basic concepts to advanced applications.

Table 2. Stages of computer vision.

Computer Vision Stage	Key Elements	Description	Related Research
Image Recognition	Image Classification, Object Recognition	Identifying and classifying objects or patterns within images.	• CNNs; • Models like ResNet and VGGNet [21].
Image Segmentation	Segmentation, Semantic Segmentation, Instance Segmentation	Partitioning images into meaningful regions.	• U-Net; • Fully Convolutional Networks (FCNs); • Mask R-CNN [19,22].
Object Detection	Bounding Boxes, Object Localization	Identifying and marking the locations of specific objects within an image.	• YOLO; • SSD [23]; • Faster R-CNN [18].
Video Analysis	Object Tracking, Action Recognition, Temporal Analysis	Analyzing movements and actions of objects in sequential frames.	• Optical Flow; • LSTM; • 3D CNN [20].

Computer vision education is generally divided into four main stages: image recognition, object detection, image segmentation, and video analysis [24]. In each stage, learners engage in hands-on activities to directly apply technologies and learn problem-solving methods. By integrating data science and computational thinking into this process, learners not only use technology but also develop logical problem-solving skills based on data. For instance, in the image recognition stage, learners can start with simple exercises using no-code platforms and gradually advance to higher-level technologies using programming languages like Python [25,26].

2.3. Computational Thinking

Computational thinking is a problem-solving approach that utilizes concepts from computer science and it has become increasingly important in modern education. It involves decomposing problems, focusing on core elements through abstraction, recognizing patterns, designing algorithms, and automating processes [8]. Computational thinking goes beyond programming. It emphasizes cultivating the ability to systematically solve complex problems.

Enhancing computational thinking in computer vision education helps students understand and apply technology effectively [27]. Decomposing problems in image or video processing and logically solving each part exemplify computational thinking. Additionally, designing algorithms and automating processes to solve computer vision problems are key elements that contribute to developing creative and logical thinking necessary for complex problem solving [28]. As shown in Table 3, the core elements of computational thinking, such as problem decomposition, abstraction, pattern recognition, algorithm design, and automation, are essential for breaking down and solving complex problems.

Table 3. Elements of computational thinking.

Element	Description and Related Research
Problem decomposition	Breaking down complex problems into smaller, manageable parts. Decomposition is essential for handling complex problems [8].
Abstraction	Focusing on important information while ignoring irrelevant details. Abstraction is the key in understanding and managing complex systems [29].

Table 3. *Cont.*

Element	Description and Related Research
Pattern recognition	Identifying recurring structures or trends to gain insights. Pattern recognition is crucial in data analysis and prediction [30].
Algorithm design	Creating step-by-step procedures to solve problems. Algorithm design is the core element of problem solving in computer science [31].
Automation	Automating repetitive tasks to increase efficiency. Automation is significantly applied in various industries.

2.4. Analysis of Previous Studies

Previous studies on computer vision education have primarily focused on deep learning algorithms and their applications. For instance, research on image recognition and object detection has concentrated on improving the performance of technologies using CNNs and YOLO models [9,18]. Ref. [32] conducted computer vision education for elementary students using deep learning-based image classification models, finding significant improvements in understanding and interest.

Studies related to computational thinking have examined the effectiveness of educational methods and tools. Ref. [33] analyzed the impact of block-based programming languages like Scratch on enhancing computational thinking, yielding positive results. However, these studies often focus on technical aspects, with relatively few addressing educational perspectives.

Educational research on effectively teaching computer vision technologies is gradually increasing, but systematic frameworks are still lacking. Particularly, research integrating data science and computational thinking into computer vision education is limited. This study aims to address these gaps by proposing a staged approach that allows learners to learn computer vision technologies by integrating neural network-based AI and symbolic AI.

3. Methods

In this study, we established a staged approach and criteria for effective teaching and learning in computer vision education. Computer vision technologies comprise multiple stages, such as image recognition, object detection, and video analysis. Developing methodologies that allow these technologies to be progressively learned in educational environments is essential. Accordingly, we proposed a four-stage educational framework that integrates neural network-based AI and symbolic AI, enabling students to systematically learn computer vision technologies. Each stage includes key elements of computer vision, such as image recognition, object detection, image segmentation, and video analysis, and is linked with various elements of data science and computational thinking.

3.1. Research Design

The research design of this study involved a systematic sequence of steps to develop and validate a staged framework for computer vision education that integrates neural network-based AI and symbolic AI, data science, and computational thinking, as shown in Figure 1.

Initially, the research objectives were clearly defined to establish the goals and scope of this study. A comprehensive literature review was then conducted to examine existing studies related to computer vision education, AI integration, data science, and computational thinking. This review helped identify gaps in current educational approaches and informed the development of the new framework.

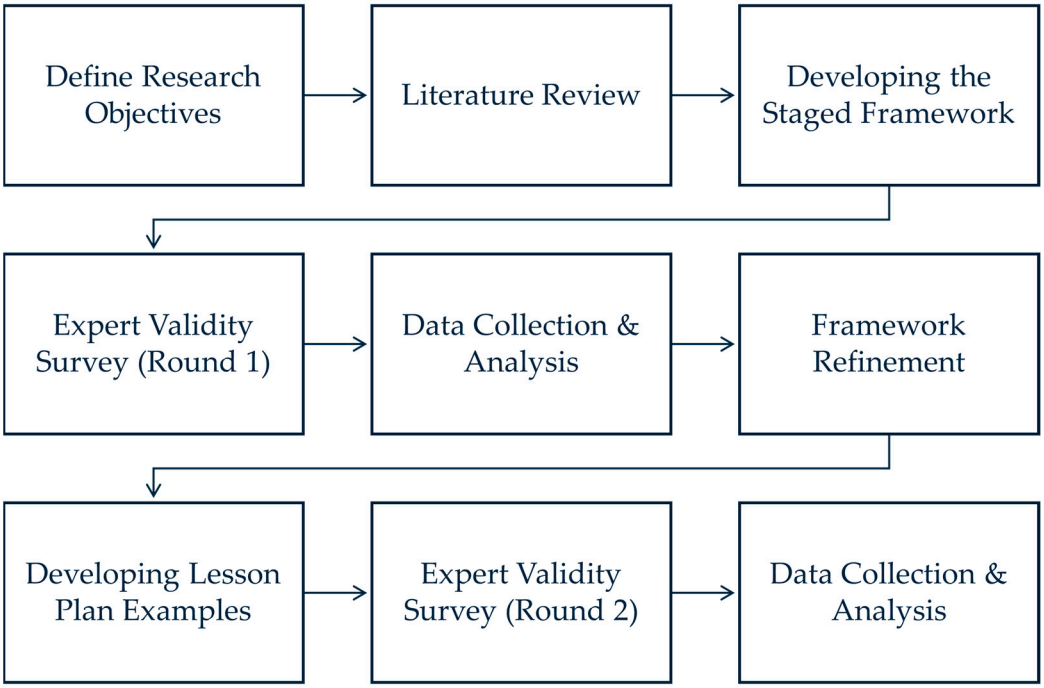


Figure 1. Research design.

Subsequently, the staged educational framework was developed, comprising four progressive levels designed to build upon each other and incorporate increasing complexity in computer vision concepts and AI methodologies. Following the development of the framework, the first round of expert validity surveys was conducted. Experts in AI education, computer vision, and curriculum development were invited to evaluate the framework using a structured questionnaire. Data collected from this survey were analyzed quantitatively and qualitatively to assess the validity and reliability of the framework.

Based on the analysis of expert feedback, necessary refinements were made to enhance the framework's validity and applicability. Detailed lesson plan examples were then developed for each stage to provide practical implementation guidance. The second round of expert validity surveys was conducted, during which experts evaluated the practical aspects of the framework and lesson plans, including the clarity of educational objectives and the feasibility of classroom implementation. Data from the second expert survey were collected and analyzed to further validate the framework and identify areas for final adjustments.

3.2. Staged Approach and Criteria for Computer Vision Education

Table 4 outlines the four-stage framework, which progressively introduces more complex computer vision concepts while integrating data science and computational thinking elements. This staged approach is based on Bloom's Taxonomy of Educational Objectives [34] and Vygotsky's Zone of Proximal Development [35], designed considering the cognitive developmental stages of learners.

Table 4. Staged framework for computer vision education.

Stage	Computer Vision Level	Computer Vision Elements	Data Science Elements	Coding Education Level	Computational Thinking Elements	Learning Activities Examples
Stage 1	Image Recognition	Image Classification, Object Recognition	Data Collection, Data Preprocessing	No-Code Platforms	Decomposition, Pattern Recognition	• Rule-based classification using decision trees or simple logic and using pre-trained models in no-code platforms. • Classifying images of lab equipment into beakers or flasks and of changes in facial expressions of mice in pain or eating sweets using rule-based criteria and pre-trained models.
Stage 2	Application of Image Recognition	Image Retrieval, Image Captioning	Data Visualization, Data Analysis	Basic Block Coding (e.g., AI blocks in Scratch 3 or mBlock V5) Basic Python Coding (e.g., OpenCV 4)	Abstraction, Algorithm Design	• Implementing conditional statements and logical operations and utilizing image features extracted by neural networks. • Explaining the usage and methods of recognized lab equipment and changes in facial expressions of mice through logical sequencing and annotations.
Stage 3	Image Segmentation and Object Detection	Segmentation, Bounding Boxes	Feature Extraction, Model Training	Advanced Block Coding (e.g., supervised learning blocks in Scratch 3 or mBlock V5) Advanced Python Coding (e.g., OpenCV 4, YOLOV8)	Automation, Simulation	• Applying rule-based criteria for object detection or segmentation and training custom neural network models. • Detecting lab equipment and facial changes of mice in experimental videos.
Stage 4	Video Analysis	Object Tracking, Action, Recognition	Model Evaluation, Model Deployment	Block Coding + Physical Computing (e.g., Scratch 3 + Arduino) Deep Learning Framework (e.g., TensorFlow, PyTorch)	Parallelization, Data Representation	• Implementing logic for action recognition and decision making and using deep learning models for complex pattern recognition. • Tracking quantitative changes in reactants and products in chemical reaction videos and changes in facial expressions of mice according to stimulus type.

In the first stage, learning focuses on image classification and object recognition. No-code platforms are utilized to help learners grasp basic computer vision concepts without prior programming experience. From the data science perspective, students learn about data collection and preprocessing, emphasizing computational thinking skills such as decomposition and pattern recognition. For example, learners can perform tasks like classifying images of laboratory equipment into categories like beakers and flasks.

In the second stage, learners study image retrieval and image captioning. Teachers have the flexibility to choose between basic block coding (e.g., AI blocks in Scratch 3 or mBlock V5) and basic Python coding (utilizing OpenCV 4) based on the learners' proficiency levels. This allows for a tailored approach that meets students where they are in their coding journey. Students acquire data visualization and data analysis techniques, strengthening abilities in abstraction and algorithm design. They can apply learning content practically through tasks such as explaining the uses and methods of the recognized laboratory equipment. This reflects the importance of accommodating different learning styles and the gradual transition from block-based to text-based coding for advancing learners' programming skills [36].

In the third stage, students learn about image segmentation and bounding box techniques. Again, teachers can select between advanced block coding (e.g., supervised learning blocks in Scratch 3 or mBlock V5) and advanced Python coding (utilizing OpenCV 4 or YOLOV8) depending on students' skills and readiness. They deepen their learning by understanding feature extraction and model training processes, focusing on enhancing automation and simulation abilities. An example task is detecting the laboratory equipment used in each stage in experimental process videos. This stage emphasizes hands-on experience with model training.

In the final stage, learners study object tracking and action recognition. Teachers have the option to choose between block coding combined with physical computing (e.g., Scratch 3 or mBlock V5 with Arduino or other hardware) and utilizing deep learning frameworks like TensorFlow 2 or PyTorch 2, based on the learners' proficiency. By combining coding and physical computing or delving into deep learning frameworks, students develop practical application skills. They learn about model evaluation and deployment processes, improving abilities in parallelization and data representation. Through tasks like tracking quantitative changes in reactants and products in chemical reaction videos, students can apply advanced computer vision technologies. This stage aligns with project-based learning approaches that enhance problem-solving skills [37].

3.3. Research Methods

To evaluate the staged approach and criteria for computer vision education, we conducted two rounds of expert validity assessments. The expert validity assessment was based on Lawshe's Content Validity Ratio (CVR) method [38], using a content validity questionnaire composed of a Likert 5-point scale. A total of 25 experts in AI education and curricula participated in evaluating the validity of this study, reviewing the validity of the educational criteria and teaching–learning activities presented in each stage.

In the first round of validity assessment, we evaluated the staged approach and criteria of computer vision education as shown in Table 5. Experts assessed whether the computer vision level presented in each stage (image recognition, object detection, image segmentation, video analysis) was appropriate from an educational perspective and whether the connections with data science and computational thinking elements were sufficient.

In the second round of validity assessment, we evaluated the validity of the teaching–learning activity examples, reflecting improvements derived from the first round. The second assessment evaluated whether the learning activities presented in each stage could effectively achieve learning objectives, the appropriateness of practical activities, and whether they could induce learner participation as shown in Table 6. Additionally, we reviewed whether the stage-specific learning activities aligned with learners' levels and

learning objectives, and whether the feedback provision methods and evaluation methods had sufficient validity.

Table 5. Validity assessment questions for the staged approach and criteria (Round 1).

Characteristic	Question
Appropriateness of Stage Division	Is the division into four stages (Image Recognition, Application of Image Recognition, Image Segmentation and Object Detection, Video Analysis) appropriate?
Validity of Computer Vision Elements	Are the computer vision elements presented at each stage (e.g., image classification, object recognition) suitable for that stage?
Connection with Data Science Elements	Are the data science elements appropriately connected with each stage of computer vision education?
Suitability of Coding Education Level	Are the coding education level presented at each stage (e.g., no-code platforms, block coding, Python coding) appropriate for learners' levels and educational goals?
Relevance of Computational Thinking Elements	Do the computational thinking elements presented at each stage relate to the learning content of that stage?
Suitability of Learning Activities Examples	Are the learning activities examples presented at each stage appropriate for achieving the learning objectives of that stage?
Connection Between Stages	Is each stage designed to progressively deepen based on the learning content of the previous stage?

Table 6. Validity assessment questions for teaching–learning activity examples (Round 2).

Characteristic	Question
Clarity of Educational Objectives	Are the educational objectives at each stage clearly set and easily understandable for learners?
Learner Engagement	Do the learning activities at each stage encourage active participation from learners?
Appropriateness of Practical Activities	Are the practical activities provided at each stage effective in achieving the learning objectives?
Provision of Feedback	Is it possible to provide appropriate feedback to learners during the learning process at each stage?
Validity of Evaluation Methods	Can the evaluation methods at each stage accurately measure learners' achievement?
Practicality of Technology Application	Are the technical elements and tools presented at each stage applicable in actual educational environments?
Connection Between Stages	Is each stage designed to progressively deepen based on the learning content of the previous stage?

And after the first expert validity assessment, we developed and provided an example of a lesson plan along with the first expert validity questionnaire as shown in Table 7. This was intended to offer concrete guidance and facilitate a more detailed evaluation of the proposed educational framework by the experts. The lesson plan example provided in Table 7, focusing on Stage 2 (utilizing image recognition) of the staged framework, demonstrates how the approach can be practically implemented in the classroom. For example, in this activity, students progress from understanding basic scientific concepts to applying advanced computer vision techniques. Students engage in hands-on experimentation, analyzing the captured images using Python, and implementing a linear regression model to predict pH values based on RGB data.

Table 7. Example of lesson plan (Stage 2: utilizing image recognition).

Grade Level	**Grade 9**	**Period**	**2 Periods (90 min)**	**Related Subjects**	**Computer Science, Science**
Learning Objectives	1. Understand and explain the concepts of image search and image captioning. 2. Create a simple image recognition program using AI blocks in block coding languages (or Python OpenCV 4). 3. Understand and apply basic concepts of data visualization and analysis.				
Phase	Teaching Activities			Learning Activities	Time (mins)
Introduction	• Introduce examples of image search and captioning in daily life. • Present learning objectives.			• Listen to the teacher's explanations and ask questions. • Share personal experiences related to computer vision technology.	10
Development	Session 1				35
	• Explain principles of image search and captioning. • Introduce AI blocks in block coding languages (or Python OpenCV 4) and explain basic usage. • Demonstrate a simple image recognition program example.			• Understand concepts and take notes. • Explore AI blocks in block coding languages (or Python OpenCV 4). • Follow along with the example program.	
	Session 2				30
	• Guide the creation of an image recognition program using AI blocks in block coding languages (or Python OpenCV 4). • Introduce data visualization tools (e.g., entry graph function or Orange3). • Guide students' practice and provide individual feedback.			• Create a program that recognizes lab equipment images and explains their uses. • Visualize recognition results in graphs. • Analyze and interpret data.	
Conclusion	• Share and provide feedback on students' results. • Summarize learning content and discuss potential developments in image recognition technology.			• Present own results and observe other students' results. • Summarize learning content and share impressions.	10
Evaluation Plan	Formative Assessment • Check understanding of image search and captioning concepts. • Observe the process of creating an image recognition program using AI blocks in block coding languages (or Python OpenCV 4). Summative Assessment • Evaluate functionality and creativity of completed image recognition programs. • Assess appropriateness of data visualization and analysis results. • Evaluate class participation and cooperative attitude.				

3.3.1. Validity Verification Method

We used Lawshe's CVR method to evaluate content validity [38]. CVR is calculated based on the results of experts' evaluations of whether each item is essential, using the following formula:

$$\mathrm{CVR} = \frac{N_e - \frac{N}{2}}{\frac{N}{2}} \quad (1)$$

where (N_e) is the number of experts indicating the item is 'valid', and (N) is the total number of experts. According to Lawshe, when 25 evaluators participate, a CVR value of 0.37 or higher indicates that the item is statistically significant. Through this method, we verified the validity of the staged educational approach and teaching–learning activities presented in this study.

3.3.2. Data Collection and Analysis

Data for this study were collected through two rounds of expert validity assessments. In the first round, we evaluated the staged approach and criteria of computer vision

education, with key evaluation items including the validity of each stage, connection with data science, and appropriateness of coding education. In the second round, we evaluated the teaching–learning activity examples for each stage, focusing on the clarity of learning objectives, appropriateness of practical activities, and the validity of feedback provision and evaluation methods.

The collected data were analyzed using SPSS Statistics 23. We calculated CVR values for each evaluation item to assess validity and incorporated improvements suggested by experts through content analysis of open-ended responses. The analysis results confirmed the validity in most items, and based on this, we revised and supplemented the staged approach and criteria for computer vision education. In particular, we received positive evaluations regarding the connection between stage-specific educational activities and data science and computational thinking elements.

3.4. Pilot Implementation

To validate the developed framework and teaching–learning process, a pilot program was conducted with 40 upper secondary school students. The entire program was delivered over four class hours as shown in Table 8.

Table 8. Pilot implementation process.

Process	Activity	
Before Implementation	• Define target. • Pre-test.	
During Implementation	Session 1	Learning scientific concepts.
	Session 2	Conduct experiment using BTB indicator.
	Session 3	Analyze pre-captured images.
	Session 4	Implement linear regression model in Python and predict pH values.
After Implementation	• Post-test. • Analyze and verify program effectiveness.	

In the first session, students learned the scientific concepts related to acid–base properties and titration using indicators. During the second session, students engaged in activities where they used the BTB indicator to distinguish between acids and bases. They observed color changes and took images of BTB solutions with pH value measurement. In the third and fourth sessions, students analyzed the RGB values of these images using a Python environment. They implemented linear regression machine learning to generate a regression model that could predict pH values based on the average RGB values of the images.

The pilot implementation aligns closely with the staged framework for computer vision education, particularly incorporating elements from the second and third stages. The activities in the second session, where students captured images of BTB solution color changes according to pH changes, correspond to the image retrieval and captioning focus of Stage 2. The third and fourth sessions, involving RGB value analysis and the implementation of a linear regression model, align with the more advanced concepts introduced in Stage 3. This includes feature extraction (RGB values) and model training (linear regression), which are central to Stage 3's learning objectives. By structuring the pilot implementation in this way, we were able to provide students with a practical, hands-on experience that progressed from utilizing image recognition to more complex data analysis and model development, mirroring the progressive nature of our staged framework.

Students completed a survey both before and after the program. The survey included questions related to their perception of computer vision education, subject knowledge, and the application of artificial intelligence. The survey questions are outlined in Table 9.

Table 9. Domains and questions for pilot implementation.

Domain	Question	Type
Perception of Computer Vision Education	1. Do you think computer vision education will be interesting?	Likert 5-point scale
	2. Do you think computer vision education will be beneficial?	
Subject Knowledge	3. Are you aware of how the color of acid-base indicators changes with pH?	
Use of AI	4. Can you predict pH values of substances using a computer vision machine learning model?	
	5. Do you understand how computer vision and AI are applied in inquiry-based activities?	

4. Results

4.1. First Expert Validity Assessment Results

The first expert validity assessment evaluated the proposed framework's structure and content, with the results summarized in Table 10.

Table 10. Results of the first expert validity assessment.

Characteristic	Question	*M*	*SD*	*CVR*
Appropriateness of Stage Division	Is the division into four stages (Image Recognition, Application of Image Recognition, Image Segmentation and Object Detection, Video Analysis) appropriate?	4.56	0.50	1.00
Validity of Computer Vision Elements	Are the computer vision elements presented at each stage (e.g., image classification, object recognition) suitable for that stage?	4.60	0.49	1.00
Connection with Data Science Elements	Are the data science elements appropriately connected with each stage of computer vision education?	4.68	0.47	1.00
Suitability of Coding Education Levels	Are the coding education levels presented at each stage (e.g., no-code platforms, block coding, Python coding) appropriate for learners' levels and educational goals?	4.64	0.48	1.00
Relevance of Computational Thinking Elements	Do the computational thinking elements presented at each stage relate to the learning content of that stage?	4.64	0.56	0.91
Suitability of Learning Activities Examples	Are the learning activities examples presented at each stage appropriate for achieving the learning objectives of that stage?	4.64	0.48	1.00
Connection Between Stages	Is each stage designed to progressively deepen based on the learning content of the previous stage?	4.72	0.45	1.00

For the 'Appropriateness of Stage Division', the mean score was 4.56 (*SD* = 0.50), indicating high validity. This suggests that the four proposed stages—image recognition, application of image recognition, image segmentation and object detection, and video analysis—are suitable for computer vision education. The 'Validity of Computer Vision Elements' received a mean score of 4.60 (*SD* = 0.49), reflecting expert consensus that the computer vision elements presented in each stage are appropriate. The 'Connection with Data Science Elements' was rated with a mean of 4.68 (*SD* = 0.47), indicating that the data science elements are well integrated with each stage of computer vision education. For the 'Suitability of Coding Education Levels', the mean score was 4.64 (*SD* = 0.48), suggesting that the coding levels—ranging from no-code platforms to Python coding—are appropriate for learners' levels and educational goals. The 'Relevance of Computational Thinking Elements' had a mean score of 4.64 (*SD* = 0.56) and a CVR of 0.91, indicating strong agreement among experts that the computational thinking elements are relevant to the learning content in each stage. The 'Suitability of Learning Activities Examples' received a mean score of 4.64 (*SD* = 0.48), confirming that the examples and activities are appropriate

for achieving the learning objectives. Lastly, the 'Connection Between Stages' had the highest mean score of 4.72 (SD = 0.45), indicating that each stage is effectively designed to build upon the previous one, facilitating the progressive deepening of knowledge and skills.

Overall, the results of the first validity assessment suggest that the stages of computer vision education are appropriately designed to match learners' levels, with meaningful connections between educational objectives and learning activities. However, some experts suggested the need for adjustments in certain stages to accommodate learners' varying levels and provide additional support where necessary.

4.2. Second Expert Validity Assessment Results

In the second expert validity assessment, we examined the validity of the teaching–learning activity examples aligned with the staged approach and criteria for computer vision education as shown in Table 11. The same 25 experts participated, evaluating each item on a Likert 5-point scale.

Table 11. Results of the second expert validity assessment.

Characteristic	Question	*M*	*SD*	*CVR*
Clarity of Educational Objectives	Are the educational objectives at each stage clearly set and easily understandable for learners?	4.52	0.50	1.00
Learner Engagement	Do the learning activities at each stage encourage active participation from learners?	4.64	0.48	1.00
Appropriateness of Practical Activities	Are the practical activities provided at each stage effective in achieving the learning objectives?	4.64	0.48	1.00
Provision of Feedback	Is it possible to provide appropriate feedback to learners during the learning process at each stage?	4.60	0.49	1.00
Validity of Evaluation Methods	Can the evaluation methods at each stage accurately measure learners' achievement?	4.68	0.47	1.00
Practicality of Technology Application	Are the technical elements and tools presented at each stage applicable in actual educational environments?	4.52	0.50	1.00

The 'Clarity of Educational Objectives' had a mean score of 4.52 (SD = 0.50), indicating that the objectives are clearly defined and understandable to learners. 'Learner Engagement' received a mean score of 4.64 (SD = 0.48), suggesting that the learning activities effectively encourage active participation. The 'Appropriateness of Practical Activities' also had a mean score of 4.64 (SD = 0.48), reflecting that the practical activities are effective in achieving the learning objectives. The 'Provision of Feedback' scored a mean of 4.60 (SD = 0.49), indicating that appropriate feedback can be provided during the learning process. 'Validity of Evaluation Methods' had a mean score of 4.68 (SD = 0.47), showing that the evaluation methods are effective in accurately measuring learners' achievement. Lastly, the 'Practicality of Technology Application' received a mean score of 4.52 (SD = 0.50), suggesting that the technical elements and tools are applicable in actual educational environments.

Overall, the second validity assessment results indicate that the teaching–learning activities proposed for each stage are suitable for achieving educational objectives and effectively engaging learners. However, some experts noted that additional learning materials might be necessary for advanced learners, implying a need for differentiated instruction.

4.3. Pilot Implementation Results

The results of the pre- and post-program surveys were analyzed using a 5-point Likert scale. Statistical analysis revealed significant improvements across all questions, as outlined in Table 12.

Table 12. Results of the pilot implementation.

Question	Category	*M*	*SD*	*t*	*p*
Perception of Computer Vision Education 1: Do you think computer vision education will be interesting?	Pre	3.5	0.877	6.345	0.000
	Post	4.575	0.636		
Perception of Computer Vision Education 2: Do you think computer vision education will be beneficial?	Pre	3.6	0.9	6.176	0.000
	Post	4.625	0.628		
Subject Knowledge 1: Are you aware of how the color of acid-base indicators changes with pH?	Pre	2.3	1.091	9.531	0.000
	Post	4.175	0.813		
Use of AI 1: Can you predict pH values of substances using a computer vision machine learning model?	Pre	2.3	0.939	8.834	0.000
	Post	4.075	0.917		
Use of AI 2: Do you understand how computer vision and AI are applied in inquiry-based activities?	Pre	1.9	1.109	5.561	0.000
	Post	3.275	1.213		

For Question 1, which assessed students' perception of the interest level of computer vision education, the mean score increased from 3.500 (*SD* = 0.877) before the program to 4.575 (*SD* = 0.636) after the program. The *t*-value of 6.345 and the *p*-value of less than 0.001 indicate a statistically significant improvement in students' interest in the subject. In Question 2, which examined the perceived usefulness of computer vision education, the mean score improved from 3.600 (*SD* = 0.900) before the program to 4.625 (*SD* = 0.628) after the program. A *t*-value of 6.176 and a *p*-value of less than 0.001 suggest that students found the program significantly more beneficial following participation. Question 3 focused on students' knowledge of the color changes of acid–base indicators. The pre-program mean score of 2.300 (*SD* = 1.091) increased significantly to 4.175 (*SD* = 0.813) after the program, with a t-value of 9.531 and a *p*-value of less than 0.001, demonstrating a notable gain in scientific understanding. Question 4, which assessed students' ability to predict pH values using a computer vision machine learning model, showed a marked improvement. The mean score rose from 2.300 (*SD* = 0.939) before the program to 4.075 (*SD* = 0.917) after the program, with a t-value of 8.834 and a *p*-value of less than 0.001, reflecting a significant enhancement in students' practical application of AI. Question 5, which evaluated students' understanding of how computer vision and AI are applied in inquiry-based activities, demonstrated an increase from a pre-program mean score of 1.900 (*SD* = 1.109) to a post-program mean score of 3.275 (*SD* = 1.213). The *t*-value of 5.561 and the *p*-value of less than 0.001 indicate a statistically significant improvement in students' comprehension of the integration of AI and computer vision in scientific exploration.

5. Discussion

In this study, we proposed a staged approach to computer vision education, exploring ways to apply elements of data science and computational thinking by integrating neural network-based AI and symbolic AI. In this discussion, we evaluate the significance and potential impact of the proposed staged framework based on the research findings and suggest implications for educational application and future research directions.

The staged approach to computer vision education was designed to enable learners to effectively grasp core concepts by integrating both neural network-based AI and symbolic AI in each learning stage. By incorporating symbolic AI, students engaged in defining explicit rules and logical criteria, which enhanced their understanding of knowledge representation and reasoning. This integrated approach allowed learners to compare symbolic reasoning with neural learning, providing a more comprehensive understanding of AI methodologies.

In the first stage—image recognition—no-code platforms were utilized to allow learners to grasp basic computer vision concepts without prior programming experience. This significantly lowered the initial barriers to learning, especially for novice learners. Learners

experienced problem-solving processes through hands-on activities and increased engagement by practically applying data collection and preprocessing—the initial stages of data science. This approach provided a crucial foundation for understanding computer vision technologies in an experimental educational environment. According to [39], no-code AI platforms can democratize AI education by making it accessible to a broader audience.

The second stage, image retrieval and captioning, was designed to let learners experience more advanced computer vision technologies. Teachers had the flexibility to choose between basic block coding (e.g., AI blocks in Scratch 3 or mBlock V5) and basic Python coding (utilizing OpenCV 4) based on the learners' proficiency levels. By using block-based programming languages, learners with limited programming experience could implement image retrieval and captioning without complex programming. Alternatively, for learners ready to advance, Python coding provided exposure to text-based programming and OpenCV 4 libraries. Through data visualization and analysis processes, they developed the ability to make better decisions using visual data. However, some experts pointed out that limitations of block-based coding might restrict the implementation of more complex functions [40]. This suggests the need for concurrent learning with text-based programming languages. Future research should develop educational strategies that organically connect block-based coding with text-based programming to facilitate a smooth transition [41].

In the third stage, object detection and image segmentation, teachers could select between advanced block coding (e.g., supervised learning blocks in Scratch 3 or mBlock V5) and advanced Python coding (utilizing OpenCV 4 or YOLOV8), depending on learners' skill levels. This approach allowed learners to directly implement object detection algorithms suitable to their programming proficiency. Using advanced block coding, learners could grasp complex concepts without being hindered by syntax, while Python coding offered a deeper dive into programming and algorithmic implementation. This process deepened learners' programming skills while encouraging active participation in data preprocessing and model training. Experts evaluated that learners could deeply understand the core concepts of computer vision by actually implementing object detection models in this stage. This was effective in cultivating practical problem-solving abilities rather than mere theoretical learning. However, some learners faced difficulties due to the challenges of text-based programming languages [36]. Additional support measures, such as scaffolding and differentiated instruction, are necessary to address this issue. Future research could propose personalized learning support that provides assistance based on individual learning levels [42].

In the fourth stage—video analysis and action recognition—teachers had the option to choose between block coding combined with physical computing (e.g., Scratch 3 or mBlock V5 with Arduino or other hardware) and utilizing deep learning frameworks like TensorFlow 2 or PyTorch 2, based on the learners' proficiency. This flexibility enabled learners to study advanced computer vision technologies through practical exercises suited to their skill levels, either by integrating coding with hardware for a tangible learning experience or by engaging with deep learning frameworks for those ready for more complex challenges. Experts assessed with high validity that learners could acquire high-level technical skills by building and analyzing actual deep learning models in this stage. However, the complexity of deep learning frameworks posed high difficulty levels for some learners [43]. This indicates the need for staged support tailored to learners' levels in future educational courses, requiring improvements in instructional design. Incorporating scaffolding techniques and providing comprehensive resources can help mitigate these challenges [44].

The pilot implementation, while focusing primarily on the second and third stages of the framework, provided valuable insights into its effectiveness. Students demonstrated significant improvements in their understanding of how computer vision and AI are applied in inquiry-based activities. This suggests that our staged approach successfully integrates computer vision concepts with practical scientific inquiry, enhancing both technical skills and subject knowledge.

The integration of neural network-based AI and symbolic AI in this staged framework is exemplified in the acid–base titration activity. In this context, neural network-based AI is represented by the linear regression model that predicts pH values based on RGB data from images of BTB indicator solutions. This model learns patterns from data without the explicit programming of rules, characteristic of neural network approaches. Concurrently, symbolic AI is incorporated through the process of analyzing RGB values from BTB indicator images using OpenCV 4 and critically examining the reasons behind the machine learning model's pH predictions. For instance, students use OpenCV 4 to extract RGB values from the indicator images, applying predefined algorithms and rules to process the visual data. This step represents a rule-based, symbolic approach to image analysis. Subsequently, students engage in a logical analysis of the machine learning model's predictions, utilizing their understanding of acid–base chemistry and color theory to interpret and validate the results. This analytical process involves creating explicit rules and heuristics to explain the relationship between RGB values and pH levels. The integration occurs as students use the symbolically derived RGB data to train the neural network model, and then apply symbolic reasoning to interpret the model's outputs.

The staged framework for computer vision education proposed in this study holds significant meaning as a new educational model integrating neural network-based AI and symbolic AI. This model helps learners gradually acquire complex concepts in computer vision and develop problem-solving skills by integrating various elements of data science and computational thinking into learning. Practice-centered learning activities provide learners with opportunities to enhance their technical abilities by solving real-world problems, offering substantial educational value. This aligns with the experiential learning theory, emphasizing learning through experience [45].

This study's findings suggest several considerations when applying the framework in educational environments. First, selecting appropriate tools and platforms that suit the educational environment and learners' characteristics is crucial. Although various tools such as no-code platforms, block-based coding, and Python were presented, effective utilization requires an individualized approach considering learners' levels and needs. The importance of tool selection in STEM education is highlighted by [46].

Additionally, support measures for enhancing educators' expertise should be prepared. There is a shortage of educators equipped with knowledge and skills in related fields such as computer vision, AI, and data science [47]. Therefore, professional development programs and the provision of educational resources for educators should be supported.

Finally, follow-up studies are needed to empirically verify the framework's effectiveness. By applying the framework in various educational environments and evaluating learners' achievement and satisfaction, the validity and utility of the framework can be confirmed. Research analyzing the long-term impact of the framework on learners' career choices or actual problem-solving abilities would also be meaningful. Longitudinal studies, as suggested by [48], could provide deeper insights into educational interventions and their lasting effects on students' understanding and application of computer vision and AI technologies.

6. Conclusions

6.1. Theoretical Contribution

This study developed a staged approach and criteria for computer vision education, proposing a new educational model by integrating neural network-based AI and symbolic AI. The findings highlight the significance of a structured educational framework that supports learners in gradually grasping complex concepts in computer vision. This aligns with the emphasis of [24] on the development of systematic methodologies in computer vision education. While existing AI education research has predominantly focused on neural network-based learning, this study integrates the logical reasoning and interpretability of symbolic AI, allowing learners to engage in deeper learning experiences. This integrated

approach offers a theoretical contribution by helping learners navigate and understand the 'black box' nature of neural networks, as discussed in [6].

By clearly demonstrating how symbolic AI is incorporated into each stage, and making this evident in the associated tables and descriptions, the contribution of symbolic AI becomes more transparent and central within the staged framework. This adjustment ensures that symbolic AI is recognized as a critical component of the learning process, aligning with the key themes and objectives of this study.

Furthermore, by incorporating elements of data science and computational thinking into the curriculum, the proposed AI education goes beyond mere technical training. It enables learners to develop problem-solving skills and derive meaningful insights during data analysis.

6.2. Practical Implications

The proposed framework provides educators with practical, stage-specific guidelines for implementing computer vision education, adaptable to various school and learner levels. The findings indicate that learning approaches involving no-code platforms and block-based programming languages are well suited for beginners, effectively reducing the initial barriers to learning programming. This approach enables learners to easily understand and practice the basic concepts of computer vision, even without prior programming knowledge. According to Ref. [49], block-based programming languages like Scratch facilitate engagement and learning among beginners.

Additionally, by participating in key stages of data science, including data collection, preprocessing, exploration, and analysis, learners can see firsthand how AI technology is used to address practical, real-world problems. The advanced learning stages utilizing text-based programming languages further enhance learners' technical capabilities and assist in cultivating practical AI implementation skills. Using practical tools like Python and OpenCV 4, learners can directly build and analyze computer vision models, thereby increasing the effectiveness of the education. Finally, practical exercises using deep learning frameworks contribute to learners acquiring advanced computer vision technologies, including real-time object tracking and action recognition.

For educators and policymakers, this study serves as a valuable reference for designing systematic, staged programs in computer vision education. Through this framework, computer vision education can be effectively conducted in diverse learning environments and serve as a practical tool to help students acquire essential AI competencies in the era of the Fourth Industrial Revolution.

6.3. Limitations and Future Research Directions

First, although a pilot test was conducted to verify the framework's effectiveness, further experimentation across various school levels is necessary to validate its general applicability. Future research should conduct experiments across different educational levels to empirically examine the general effectiveness and long-term impact of the framework on learners' academic performance and overall educational outcomes. Such studies would help in understanding practical differences in outcomes and identifying potential areas for further refinement of the framework.

Second, the current study's validation relied primarily on theoretical foundations and expert assessments, without delving into personalized educational approaches that consider individual learner characteristics, prior knowledge, and learning preferences. Future studies should develop customized learning strategies that cater to individual learner differences and propose AI educational methodologies based on these strategies. In particular, educational models that provide personalized feedback in each stage according to the learner's level are necessary, aligning with the principles of adaptive learning [42].

Third, this research concentrated on computer vision education at the elementary and secondary levels. However, computer vision is also highly relevant in higher education and vocational training. Subsequent research should develop computer vision education

frameworks targeting university students or vocational trainees and explore educational methods suitable for their specific characteristics and needs.

Lastly, while the proposed framework emphasizes technical aspects, the utilization of computer vision technology entails ethical and social considerations. Concerns such as privacy in facial recognition and liability in autonomous systems point to the necessity of incorporating ethical education into the curriculum. Future research should integrate ethical perspectives into computer vision education, enabling learners to critically reflect on the societal impacts of technology, as recommended by Ref. [50].

Author Contributions: Conceptualization, I.-S.J. and S.-J.K.; methodology, S.J.K. and S.-J.K.; validation, I.-S.J., S.J.K., and S.-J.K.; formal analysis, I.-S.J.; investigation, I.-S.J.; resources, S.J.K. and S.-J.K.; data curation, I.-S.J.; writing—original draft preparation, I.-S.J.; writing—review and editing, S.J.K. and S.-J.K.; visualization, I.-S.J.; supervision, S.-J.K.; project administration, S.-J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board of Korea National University of Education.

Informed Consent Statement: Informed consent was obtained from all subjects involved in this study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Canedo, D.; Neves, A.J. Facial expression recognition using computer vision: A systematic review. *Appl. Sci.* **2019**, *9*, 4678. [CrossRef]
2. Ngoc Anh, B.; Tung Son, N.; Truong Lam, P.; Phuong Chi, L.; Huu Tuan, N.; Cong Dat, N.; Huu Trung, N.; Umar Aftab, M.; Van Dinh, T. A computer-vision based application for student behavior monitoring in classroom. *Appl. Sci.* **2019**, *9*, 4729. [CrossRef]
3. Wu, M.; Li, C.; Yao, Z. Deep active learning for computer vision tasks: Methodologies, applications, and challenges. *Appl. Sci.* **2022**, *12*, 8103. [CrossRef]
4. Russell, S.J.; Norvig, P. *Artificial Intelligence: A Modern Approach*; Pearson: London, UK, 2016.
5. Brachman, R.J. *Knowledge Representation and Reasoning*; Morgan Kaufman: Burlington, MA, USA; Elsevier: Amsterdam, The Netherlands, 2004.
6. Marcus, G.; Davis, E. *Rebooting AI: Building Artificial Intelligence We Can Trust*; Vintage: New York, NY, USA, 2019.
7. Provost, F.; Fawcett, T. Data science and its relationship to big data and data-driven decision making. *Big Data* **2013**, *1*, 51–59. [CrossRef]
8. Wing, J.M. Computational thinking. *Commun. ACM* **2006**, *49*, 33–35. [CrossRef]
9. Bengio, Y.; Goodfellow, I.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2017; Volume 1.
10. Jeon, I.-S.; Kim, S.-Y.; Kang, S.-J. Developing Standards for Educational Datasets by School Level: A Framework for Sustainable K-12 Education. *Sustainability* **2024**, *16*, 4954. [CrossRef]
11. Kim, S.-Y.; Jeon, I.; Kang, S.-J. Integrating Data Science and Machine Learning to Chemistry Education: Predicting Classification and Boiling Point of Compounds. *J. Chem. Educ.* **2024**, *101*, 1771–1776. [CrossRef]
12. Jagadish, H.V.; Gehrke, J.; Labrinidis, A.; Papakonstantinou, Y.; Patel, J.M.; Ramakrishnan, R.; Shahabi, C. Big data and its technical challenges. *Commun. ACM* **2014**, *57*, 86–94. [CrossRef]
13. García, S.; Luengo, J.; Herrera, F. *Data Preprocessing in Data Mining*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 72.
14. Tukey, J.W. *Exploratory Data Analysis*; Reading/Addison-Wesley: Hoboken, NJ, USA, 1977.
15. Hastie, T.; Tibshirani, R.; Friedman, J.H.; Friedman, J.H. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 2.
16. Arlot, S.; Celisse, A. A survey of cross-validation procedures for model selection. *Stat. Surv.* **2010**, *4*, 40–79. [CrossRef]
17. Sacha, D.; Sedlmair, M.; Zhang, L.; Lee, J.; Weiskopf, D.; North, S.; Keim, D. Human-centered machine learning through interactive visualization. In Proceedings of the 24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, 27–29 April 2016; pp. 641–646.
18. Redmon, J. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
19. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; pp. 234–241.

20. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 1–9.
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
22. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
23. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
24. Szeliski, R. *Computer Vision: Algorithms and Applications*; Springer Nature: Berlin/Heidelberg, Germany, 2022.
25. Kang, M.; Yul, P.S.; Chung, M.-a.; Han, D.-h. *No-Code Ai: Concepts and Applications in Machine Learning, Visualization, and Cloud Platforms*; World Scientific Publishing Company: Singapore, 2024.
26. Kling, N.; Runte, C.; Kabiraj, S.; Schumann, C.-A. Harnessing sustainable development in image recognition through no-code AI applications: A comparative analysis. In Proceedings of the International Conference on Recent Trends in Image Processing and Pattern Recognition, Msida, Malta, 8–10 December 2021; pp. 146–155.
27. Prottsman, K. *Computational Thinking Meets Student Learning: Extending the ISTE Standards*; International Society for Technology in Education: Washington, DC, USA, 2022.
28. Grover, S.; Pea, R. Computational thinking in K–12: A review of the state of the field. *Educ. Res.* **2013**, *42*, 38–43. [CrossRef]
29. Aho, A.V. Computation and computational thinking. *Comput. J.* **2012**, *55*, 832–835. [CrossRef]
30. Shute, V.J.; Sun, C.; Asbell-Clarke, J. Demystifying computational thinking. *Educ. Res. Rev.* **2017**, *22*, 142–158. [CrossRef]
31. Barr, D.; Harrison, J.; Conery, L. Computational thinking: A digital age skill for everyone. *Learn. Lead. Technol.* **2011**, *38*, 20–23.
32. Budiman, S.N.; Lestanti, S.; Rahmat, M.F.; Nafis, M. Development of Computer Vision-Based Sciences Educational Games for Elementary Schools. *J. Inform. Inf. Syst. Softw. Eng. Appl. (INISTA)* **2023**, *6*, 21–30.
33. Brennan, K.; Resnick, M. New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 Annual Meeting of the American Educational Research Association, Vancouver, BC, Canada, 13–17 April 2012; p. 25.
34. Bloom, B.S.; Krathwohl, D.R.; Masia, B.B. *Taxonomy of Educational Objectives-Handbook II: Affective Domain*; David McKay: New York, NY, USA, 1964.
35. Vygotsky, L.S. *Mind in Society: The Development of Higher Psychological Processes*; Harvard University Press: Cambridge, MA, USA, 1978; Volume 86.
36. Weintrop, D.; Wilensky, U. Transitioning from introductory block-based and text-based environments to professional programming languages in high school computer science classrooms. *Comput. Educ.* **2019**, *142*, 103646. [CrossRef]
37. Krajcik, J.S.; Blumenfeld, P.C. *Project-Based Learning*; The Cambridge Handbook of the Learning Sciences; Cambridge University Press: Cambridge, UK, 2005; pp. 317–334.
38. Lawshe, C.H. A quantitative approach to content validity. *Pers. Psychol.* **1975**, *28*, 563–575. [CrossRef]
39. Deshpande, M. No-Code AI: Empowering Business Users to Harness the Power of Artificial Intelligence. *J. Artif. Intell. Mach. Learn. Data Sci.* **2024**, *2*, 354–359. [CrossRef]
40. Weintrop, D.; Wilensky, U. To block or not to block, that is the question: Students' perceptions of blocks-based programming. In Proceedings of the 14th International Conference on Interaction Design and Children, Medford, MA, USA, 21–25 June 2015; pp. 199–208.
41. Grover, S.; Basu, S. Measuring student learning in introductory block-based programming: Examining misconceptions of loops, variables, and boolean logic. In Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education, Seattle, WA, USA, 8–11 March 2017; pp. 267–272.
42. Woolf, B.P. *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning*; Morgan Kaufmann: Burlington, MA, USA, 2010.
43. Deng, L.; Yu, D. Deep learning: Methods and applications. *Found. Trends Signal Process.* **2014**, *7*, 197–387. [CrossRef]
44. Reiser, B.J. Scaffolding complex learning: The mechanisms of structuring and problematizing student work. In *Scaffolding*; Psychology Press: Hove, UK, 2018; pp. 273–304.
45. Kolb, D.A. *Experiential Learning: Experience as the Source of Learning and Development*; FT Press: Upper Saddle River, NJ, USA, 2014.
46. Honey, M.; Pearson, G.; Schweingruber, H.A. *STEM Integration in K-12 Education: Status, Prospects, and an Agenda for Research*; National Academies Press: Washington, DC, USA, 2014; Volume 500.
47. Yadav, A.; Gretter, S.; Hambrusch, S.; Sands, P. Expanding computer science education in schools: Understanding teacher experiences and challenges. *Comput. Sci. Educ.* **2016**, *26*, 235–254. [CrossRef]
48. Wagner, E.; Ice, P. Data changes everything: Delivering on the promise of learning analytics in higher education. *Educ. Rev.* **2012**, *47*, 32.
49. Resnick, M.; Maloney, J.; Monroy-Hernández, A.; Rusk, N.; Eastmond, E.; Brennan, K.; Millner, A.; Rosenbaum, E.; Silver, J.; Silverman, B. Scratch: Programming for all. *Commun. ACM* **2009**, *52*, 60–67. [CrossRef]
50. Mittelstadt, B.D.; Allo, P.; Taddeo, M.; Wachter, S.; Floridi, L. The ethics of algorithms: Mapping the debate. *Big Data Soc.* **2016**, *3*, 2053951716679679. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.