applied sciences

MDPI

Article

Voice-Controlled Robotics in Early Education: Implementing and Validating Child-Directed Interactions Using a Collaborative Robot and Artificial Intelligence

Cristhian A. Aguilera [1,*], Angela Castro [2], Cristhian Aguilera [3] and Bogdan Raducanu [4,5]

[1] Facultad de Ingeniería, Arquitectura y Diseño, Universidad San Sebastián, Lago Panguipulli 1390, Puerto Montt 5501842, Chile
[2] Facultad de Educación, Universidad San Sebastián, Lago Panguipulli 1390, Puerto Montt 5501842, Chile; angela.castro@uss.cl
[3] Departamento de Ingeniería Eléctrica y Electrónica, Facultad de Ingeniería, Universidad del Bío-Bío, Concepción 4051381, Chile; cristhia@ubiobio.cl
[4] Computer Vision Center, Edifici O, Campus UAB, 08193 Bellaterra, Spain; bogdan@cvc.aub.es
[5] Computer Science Department, Universitat Autònoma de Barcelona, Campus UAB, 08193 Bellaterra, Spain
* Correspondence: cristhian.aguilera@uss.cl

Abstract: This article introduces a voice-controlled robotic system for early education, enabling children as young as four to interact with robots using natural voice commands. Recognizing the challenges posed by programming languages and robot theory for young learners, this study leverages recent advancements in artificial intelligence, such as large language models, to make robots more intelligent and easier to use. This innovative approach fosters a natural and intuitive interaction between the child and the robot, effectively removing barriers to access and expanding the educational possibilities of robotics in the classroom. In this context, a software pipeline is proposed that translates voice commands into robot actions. Each component is tested using different deep learning models and cloud services to determine their suitability, with the best ones being selected. Finally, the chosen setup is validated through an integration test involving children aged 4 to 6 years. Preliminary results demonstrate the system's capability to accurately recognize and execute voice commands, highlighting its potential as a valuable educational tool for early education.

Keywords: cobots; education; human–robot interaction; artificial intelligence

Citation: Aguilera, C.A.; Castro, A.; Aguilera, C.; Raducanu, B. Voice-Controlled Robotics in Early Education: Implementing and Validating Child-Directed Interactions Using a Collaborative Robot and Artificial Intelligence. *Appl. Sci.* **2024**, *14*, 2408. https://doi.org/10.3390/app14062408

Academic Editors: Eloy López Meneses, Carlos Hervás-Gómez, María Dolores Díaz-Noguera and Pedro Román-Graván

Received: 26 February 2024
Revised: 10 March 2024
Accepted: 10 March 2024
Published: 13 March 2024

1. Introduction

The application of robotics within educational settings has emerged as a transformative force, promising to redefine the pedagogical landscape for the 21st-century learner [1]. Integrating robotics in early education fosters technological literacy and cultivates essential skills such as creativity, collaboration, and problem-solving, all pivotal within the STEM domain [2]. However, the challenge of executing complex tasks with robotics using programming languages and kits can be overwhelming for young learners. This often restricts the scope of their achievable projects, potentially restricting their full creative and intellectual development.

Current educational robotic kits, while introducing technology and engineering concepts, primarily offer limited and predefined interaction paradigms, restricting the scope of tasks that can be explored and executed by preschool-aged children [3,4]. These constraints arise not from a lack of capability on the part of the children but from the inadequacy of the robots and programming languages designed for their use. Complex concepts like control theory, computer vision, and machine learning are still out of reach for preschool-aged children because the programming and concepts involved are too advanced.

This study suggests that by simplifying the user interfaces and enhancing the intelligent response capabilities of robots, we can significantly broaden the range of conceptual

and practical skills accessible to young learners, even four-year-olds. With recent advancements in artificial intelligence (e.g., [5,6]), computer vision (e.g., [7,8]), and robotics (e.g., [9,10]), there lies an opportunity to innovate the methods of interaction between children and educational robots [11]. Evidence from recent literature indicates that even preschoolers can develop an understanding of complex scientific and mathematical concepts when provided with intuitive and engaging tools [12,13].

Addressing this opportunity, our research introduces a voice-controlled robotic system designed for early education settings. The system integrates advanced technologies, including speech-to-text (STT), large language models (LLMs), and object detection, enabling it to understand and respond to verbal commands from children as young as four. This capability underscores the potential for incorporating sophisticated robotics into early childhood educational environments. Through this implementation, we demonstrate that collaborative robotics, typically reserved for industrial settings (e.g., [14,15]), are equally effective and engaging in early education, fostering a comprehensive approach that encourages young learners to explore and understand complex concepts across various knowledge domain.

The primary contributions of this work are as follows:

- We present a proof of concept for a robotic system responsive to children's verbal instructions. This system not only advances the field of educational robotics with its novel capabilities but also serves as a foundational platform for further empirical exploration within the domain.
- Our comprehensive evaluation of critical technologies, including speech-to-text, large language models, and object detection models, to process and interpret children's instructions, highlights these technologies' potential to enhance educational experiences and pinpoints challenges for future research.
- To the best of our knowledge, this research represents the first attempt to utilize collaborative robots, similar to those in industrial settings, in both suitable and beneficial ways for early childhood education. This endeavor marks a significant step forward in educational robotics and opens new opportunities for development within educational contexts.

2. System Architecture and Methods

2.1. Experimental Setup: An Overview

Our robot system's experimental setup enables dynamic interactions with children, as Figure 1 shows. A robot, central to this setup, operates within a clearly defined workspace marked by four QR codes. These codes identify the area where the system detects and manipulates objects. While the workspace serves as the main stage for object detection and interaction, a key feature of our system enables the robot to move objects beyond this area into designated color boxes. These color boxes act as destinations for the objects children need to relocate as part of their tasks during the system's experimental evaluation.

The robot's interaction with children follows a structured sequence of steps, as depicted in Figure 2. The process begins with a child using a wake-up word that is easy to pronounce and recognizable by the voice recognition system. This activation triggers the robot to expect a new instruction. Children then have six seconds to issue voice commands for tasks such as moving to a specific object, moving the robot arm down or up, and grabbing or dropping an object. A speech-to-text system converts these voice commands into written text.

Next, an LLM analyzes the written text to determine the child's desired action (semantic analysis). An object detection system then detects and localizes objects within the scene and relays this information to a robot action module. This module combines the action information from the LLM system with the object localization data to execute the given function.

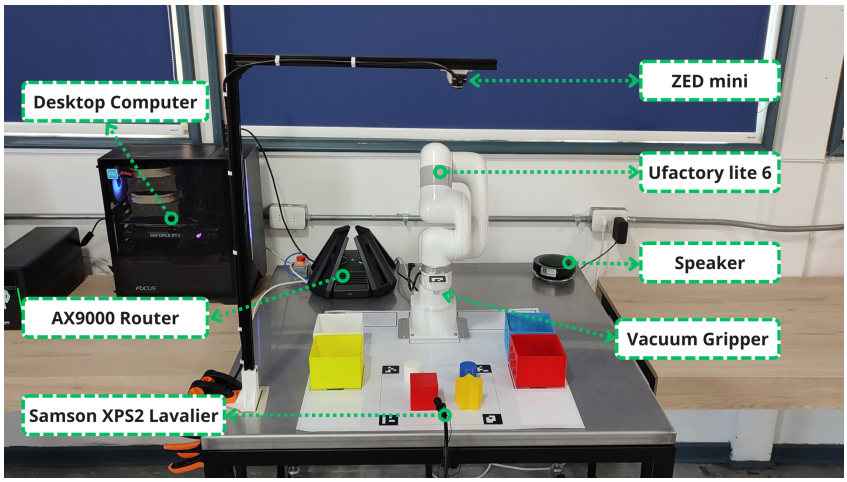


Figure 1. Detailed view of the experimental setup, showcasing its various components.

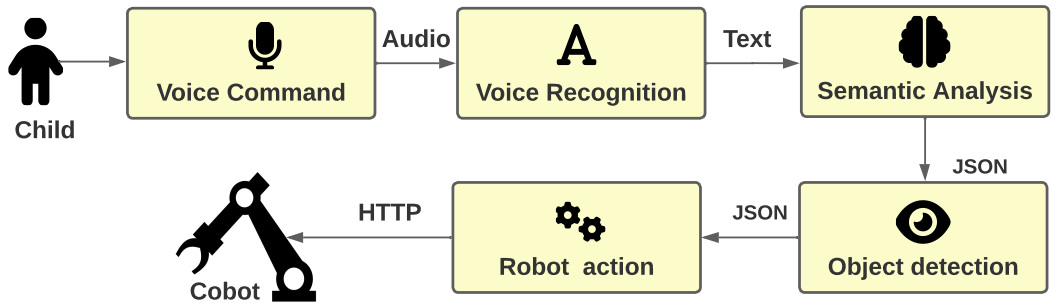


Figure 2. Flowchart detailing the child–robot interaction sequence, from voice command to the robot's action execution. The interaction initiates with a child's voice command, recorded by a microphone, and converted into text by a STT system. This text is analyzed by an LLM to determine the necessary action. Simultaneously, an object detection module gathers information on the environment's elements and their positions, essential for executing the action. This information, combined with the LLM's analysis, enables the robot action module to guide the robot's movements accurately.

If a command is unclear or the action is not feasible, the robot responds, *I do not understand the instruction; try again*, prompting the child to rephrase their request more clearly. This structured interaction ensures the robot responds appropriately to children's commands and facilitates effective communication and debugging.

2.2. System Setup Details

The system primarily comprises a collaborative robot arm, the Ufactory Lite 6, with a vacuum gripper to manipulate objects. A ZED Mini camera (Stereo Labs, San Francisco, CA, USA), mounted on an aluminum frame, monitors the robot's workspace for scene observation, distance measurement, and object detection. The Samson XPS2 microphone captures (Samson AG, Frankfurt, Germany) voice commands, allowing user interaction with the robot. The processing unit is a PC equipped with an NVIDIA RTX3080 Ti graphics card (NVIDIA, Santa Clara, CA, USA), capable of running deep learning models. The robot and computer communicate via high-speed HTTP messages over an AX9000 router (Xiaomi, Pekin, China).

2.3. Implementation Details

This section offers a concise overview of the core technologies and methodologies used in the implementation, tracing the sequence from voice recognition to robot actions. It highlights the key technologies driving each process and explains the rationale behind their selection while adhering to strict safety and privacy guidelines.

2.3.1. Voice Recognition Module

Understanding voice instructions from children involves a three-step process. The first step is to recognize when a child is about to give an instruction, signaled by a wake-up word. This wake-up word is a specific term that the system continuously monitors for, alerting it to the start of the interaction. For this purpose, we utilize Picovoice [16], a renowned web API that has been recognized for its proficiency in multilingual voice recognition capabilities, ensuring reliable detection across various languages.

Once the wake-up word is detected, a six-second window is provided for the child to deliver their instruction. This duration was carefully chosen based on our observations; initially, we allocated four seconds, but this proved insufficient, especially for 4-year-olds. Six seconds offers a more comfortable time frame for children to articulate their instructions.

The spoken message is captured and forwarded to an STT system for transcription in the subsequent step. This system converts the audio input into text, which the semantic analysis module will then process to discern the child's intention. For the STT system, we tested various deep learning models and web services, including Whisper Base [6], Whisper Tiny [6], Whisper Small [6], Whisper Medium [6], Whisper Large [6], Vosk Small [17], Vosk Normal [17], and Google STT [18] with contextual information, as well as Leopard [16]. Ultimately, we opted for Google's STT service, which uses contextual information, i.e., providing hints of the expected words.

For a detailed comparison of the various STT models and services evaluated in this work, please refer to the results section. The result section provides an in-depth analysis of the effectiveness and suitability of different models and services in converting children's voice instructions to text.

2.3.2. Semantic Analysis Module

After the system transcribes the child's voice command, it tasks an LLM system with interpreting the message and reforming it into a structured dictionary with five fields: Result, Action, Object, Color, and Size. The Result field indicates the success of the interpretation, returning true for success and false otherwise. Action specifies the robot's required action, encompassing six possibilities: go to a bucket or object, go down, go up, grab, release an object, and restart (an internal command). The Object field identifies which object the action targets. Color and Size are optional, and dependent on the scene's objects. For instance, Color becomes critical when differentiating between multiple cube objects of varying colors. Similarly, the need for Size arises under similar circumstances. We intentionally avoid test scenes with objects of similar color and size to prevent ambiguity in our experiments.

The prompt was the following: Interpret the voice command from a child, converted to text, to control a robot (Little hand). Respond with Result, Action, Object, Color, and Size. If the color or size is not specified, respond with not specified. If it was impossible to understand the phrase, respond with Result: error. Otherwise, respond with Result: ok. Possible actions include go down, go up, release, grab, go to, and restart. Possible colors are red, white, blue, and yellow. Possible sizes are large and small. Possible objects are star, cylinder, cube, and box; any other object is considered an error. The voice command is: {**STT transcript**} Note: Commands may include variations in expression and common language errors made by children. Normalize the response to the established categories. The output is a JSON file with the following fields: result, action, object, color, and size. Additionally, children might use terms like grab, suck, suction, or similar to indicate the action of grabbing an object.

The previously described prompt is a translation of the prompt used in our experiments, originally in Spanish. {**STT transcript**} represents the output from the STT module, which converts the child's spoken voice command into written text. For example, an STT transcript might be as follows: "Move the robot to the blue cube".

The output is a JSON file containing the fields previously described (see Figure 3). In the results section, a thorough evaluation is conducted comparing the Gemini [19] LLM

service with ChatGPT [20] versions 3.5 and 4. We ultimately chose Gemini due to its superior runtime speed and performance. We did not test open-source LLMs locally due to the need for sufficiently powerful GPUs to run these models.

```
{
  ‘‘Result’’: ‘‘ok’’,
  ‘‘Action’’: ‘‘go to’’,
  ‘‘Object’’: ‘‘cylinder’’,
  ‘‘Color’’: ‘‘white’’,
  ‘‘Size’’: ‘‘unspecified’’
}
```

Figure 3. Example JSON output for interpreting the voice command, “go to the white cylinder”.

2.3.3. Vision-Based Object Detection Module

The object detection module, crucial for the system, finds and classifies objects within the robot’s QR code-defined workspace (Figure 1). These QR codes serve to avoid false detections outside the designated area. The detector’s responsibility is to identify simple geometric shapes, such as cylinders, cubes, and stars, in four distinct colors: yellow, white, blue, and red. These particular shapes and colors are familiar to children within the targeted age group of the study. Additionally, their inclusion intentionally supports educational objectives, including teaching children to categorize objects based on various attributes.

From a technical perspective, an object detector scans an image to find a list of pre-trained objects. It returns each detected object’s label and its bounding box coordinates, defined by four points: (x1, y1) for the upper left and (x2, y2) for the bottom right corners. Two main types of learning-based object detection methods exist: one-stage and two-stage. One-stage methods, including YOLO [8] and SSD [21], tend to be faster but less accurate than two-stage methods such as Cascade R-CNN [22] and Mask R-CNN [23], as mentioned in [24]. In this work, after an empirical evaluation described in Section 3.2, we decided to use YOLOv8N, an advanced version of the popular one-stage method in the YOLO family of models (e.g., [8,25,26]). YOLOv8N achieves a good balance between runtime speed and accuracy.

To ensure the robot can accurately grasp objects detected by this module, the locations of these objects must be expressed in terms of the robot’s coordinate system, a process achieved through hand-to-eye calibration [27]. This involves attaching a QR code to the robot’s end effector and moving it through known positions within the camera’s view to compute a transformation matrix that aligns the camera’s coordinate system with the robot’s, significantly simplified by the ZED Mini camera’s point cloud capability. This matrix converts detected object locations into robot coordinates, facilitating precise positioning of the robot’s end effector for object manipulation.

2.4. Safety and Privacy Measures

Although collaborative robots (cobots) are designed with inherent safety features for human interaction, we implemented additional precautions in our setup. Specifically, we established a designated safety zone for interactions between the robot and children. Moreover, children were always supervised during the process to ensure utmost safety.

Regarding privacy, we secured written consent from the parents before participation, ensuring that any data collected would be utilized exclusively for academic research purposes and in strict adherence to privacy and data protection laws currently in effect in Chile.

3. Results

3.1. Datasets

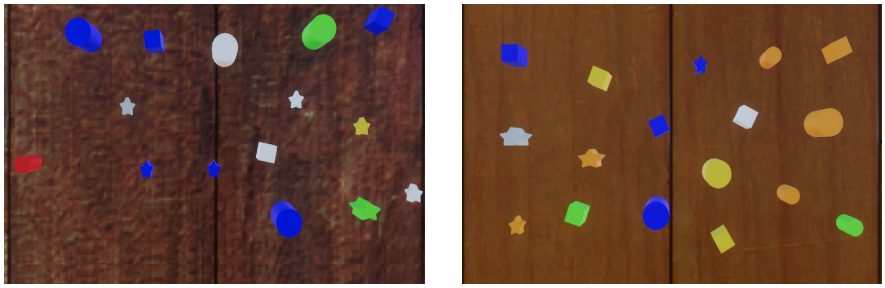
For this experiment, we generated two types of datasets created for two primary purposes: (i) to train and evaluate the suitability of various deep learning models and web

services implemented in our project and (ii) to elucidate the rationale and selection process for the tools employed.

3.1.1. Geometric Figures Dataset

We generated a synthetic dataset (SGFD) to train object detectors that would later be used to identify objects within the robot's workspace. This dataset was created using the Unity game engine and its Perception package [28], primarily for training purposes. It facilitated the generation of abundant training data featuring squares, cylinders, and stars in various colors, mirroring those used during the system's evaluation with children. Given that the actual objects were produced with a 3D printer and placed against a white background, we anticipated a minimal gap between the synthetic and real-world environments. A random object generator was employed to produce a variety of shapes and colors distributed evenly across different backgrounds for robustness. The dataset's composition, including the number of images and instances, is detailed in Table 1, with sample images shown in Figure 4.

We also captured and labeled 100 image samples from the workspace using the ZED Mini camera and 3D-printed pieces to evaluate their applicability in real-world scenarios and verify our hypothesis regarding the minimal synthetic-to-real gap. This collection has been named the real geometric figures dataset (RGFD). Like the SGFD, this real dataset includes the same three classes: cylinder, star, and cube. Specifically, it comprises 400 object instances, with 161 instances of cubes, 89 cylinders, and 150 stars.



(**a**) sample 1 (**b**) sample 2

Figure 4. Image samples from the synthetic dataset used to train the object detector model.

Table 1. Distribution of images across training, validation, and testing subsets within the synthetic dataset, which includes three classes: cylinder, star, and cube. The dataset comprises 24,568 instances of cubes, 24,064 instances of cylinders, and 24,098 instances of stars.

Subset	Number of Images
Training	3500
Validation	500
Testing	1001
Total	5001

3.1.2. Voice Instructions Dataset (VID)

To assess the performance of different STT models and web services, we collected audio recordings featuring voice commands from children aged 4 to 8, all in Spanish, the language used in all our experiments. These recordings included a variety of commands, such as *move to the white cylinder*, *go down*, and *go up*, among others, aimed at controlling the robot to perform a wide range of tasks. We gathered 311 audio recordings, creating a substantial dataset for evaluating the robot's task execution capabilities. This set is named VID. The recordings exhibit varying sound qualities, as we requested parents to use their smartphones for the recordings, resulting in a diverse range of audio qualities within the dataset.

Observing that numerous recordings in the original dataset commenced directly with a child's voice—an approach generally less favorable for STT systems—we created an alternative version (VID1S). In this version, we added 1 s of silence at the start of each recording. This adjustment was made to explore whether introducing a brief period of silence could enhance the STT system's performance, particularly considering the abrupt beginning of most of the original recordings.

3.2. Object Detection Results

We trained four different models for object detection in our robot. These models are YOLOv8N [29], YOLOv8M [29], YOLOv8X [29], and RT-DETR-L [7]. All models were pre-trained on the COCO dataset [30]. The objective was to evaluate the capabilities of various models, each with a different number of parameters, to find the optimal balance between accuracy and runtime, making the application suitable for the experiment. The models were solely trained on the synthetic dataset (SGFD) and tested on the real dataset (RGFD). The training utilized default hyperparameters, including various augmentation techniques such as image rotation, cropping, and color jitter. We conducted the training and evaluation on an NVIDIA RTX 3080 Ti, with all models undergoing 100 training epochs.

We used the standard metric mean average precision (mAP) to evaluate the trained detector models' accuracy. The mAP evaluates object detection models by considering both precision (the model's accuracy in identifying only relevant objects) and recall (the model's ability to identify all relevant objects). It is calculated across various IoU (intersection over union) thresholds. IoU is a measure used to quantify the overlap between the predicted and ground truth bounding boxes.

Table 2 presents the results of two popular mAP variants, mAP50 and mAP50-95, demonstrating that all models could reliably identify objects in the real dataset, even without being trained on real-world data. This success was anticipated, considering the toy-like setup of our experiment does not significantly challenge modern object detection models, thus simplifying the task. Among the models, YOLOv8N emerged as the most suitable choice because of its optimal balance between runtime efficiency and accuracy. Notably, our approach did not rely on the object detection models for color recognition; instead, we employed a straightforward color distance method. Specifically, we calculated the average pixel value within the detected object's bounding box and compared it to a reference value. This strategy effectively utilized the clear differentiation of colors in the RGB spectrum to accurately identify color ranges. An intriguing observation was that the transformer-based model, RT-DETR, was the only one that underperformed, seemingly unable to generalize effectively between the synthetic and the real dataset.

Figure 5 displays a snapshot taken by the camera alongside the outcomes following the object detection process.

Table 2. Performance evaluation of object detection models on RGDF. The table displays the count of instances identified within the 100 images that comprise the dataset.

Model	Class	Instances	mAP50	mAP50-95	GFLOPs
YOLOv8N	all	400	0.994	0.834	8.1
	cube	161	0.993	0.836	
	cylinder	89	0.995	0.882	
	star	150	0.995	0.783	
YOLOv8M	all	400	0.983	0.818	78.7
	cube	161	0.970	0.777	
	cylinder	89	0.995	0.881	
	star	150	0.983	0.795	

Table 2. *Cont.*

Model	Class	Instances	mAP50	mAP50-95	GFLOPs
YOLOv8X	all	400	0.983	0.812	257.4
	cube	161	0.959	0.768	
	cylinder	89	0.995	0.862	
	star	150	0.995	0.805	
RT-DETR	all	400	0.776	0.501	103.4
	cube	161	0.819	0.497	
	cylinder	89	0.951	0.632	
	star	150	0.558	0.372	

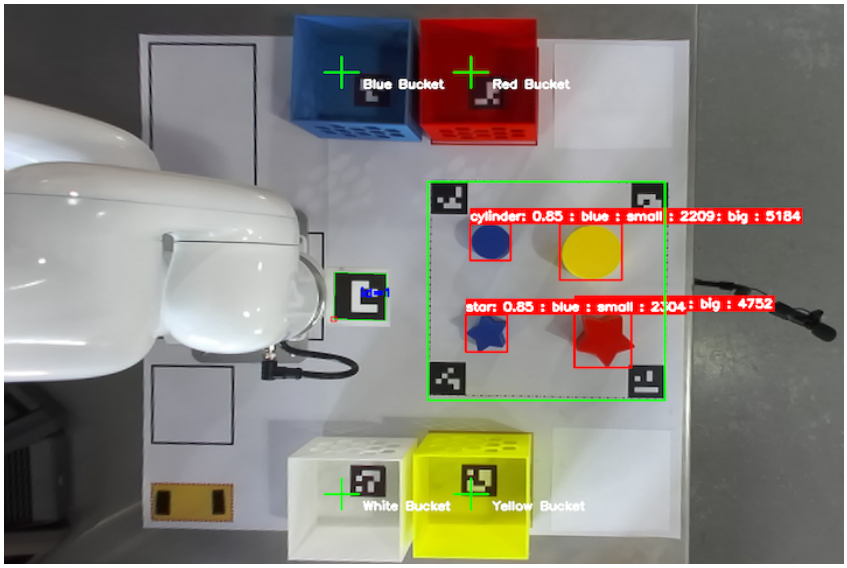


Figure 5. Top view of the working area with objects detected using YOLOv8N.

3.3. Speech-to-Text Results

The availability of STT models has increased significantly in recent years, ranging from free, open-source models to cloud-based services. Although standard benchmarks exist to evaluate their performance, assessing specific requirements that differ from the conditions of available benchmarks is often necessary. This consideration is particularly relevant when working with children aged four who are non-English speakers, in this case, Spanish speakers. Therefore, various models and services were evaluated using the dataset we captured to determine which model performs better regarding the word error rate (WER). WER is a popular metric used by automatic speech recognition systems. It tells you how many mistakes a system makes in recognizing spoken words. A lower WER indicates a more accurate system with fewer word recognition mistakes.

From Table 3, it is evident that STT systems face challenges when processing voice instructions from small children. This difficulty can be attributed to various factors. One factor is the language used (Spanish); another is the limited amount of data available for young children compared to adults, leading to generally poorer STT performance in this age group. Additionally, speech development in young children is a critical factor. For instance, children in Spanish-speaking countries often struggle with words containing the *R* sound. The comparison reveals that the Whisper Large model and Google STT using context performed the best. However, their performance still falls short of the results obtained in standard benchmarks, highlighting the complexities involved in working with children's speech.

Table 3. WER comparison across various STT models. Here, WB = Whisper Base, WT = Whisper Tiny, WS = Whisper Small, WM = Whisper Medium, WL = Whisper Large, VS = Vosk Small, VN = Vosk Normal, G = Google STT, GC = Google STT using context, L = Leopard.

Dataset	WB [6]	WT [6]	WS [6]	WM [6]	WL [6]	VS [17]	VN [17]	G [18]	GC [18]	L [16]
VID	1.20	1.47	0.67	0.90	0.37	0.64	0.42	0.45	0.33	0.73
VID1S	0.87	0.97	0.82	0.64	0.39	0.68	0.44	0.44	0.26	0.66

Figure 6 illustrates the relationship between runtime and WER. Although the Whisper Large model offers favorable WER performance, its slower processing speed may not be practical for real-time applications, especially in contexts where children are involved and quick interaction is crucial. The Vosk model presents a balanced trade-off between speed and accuracy, but it does not match the accuracy of Google's context-aware STT, which outperforms other models in both speed and precision. Additionally, it is advantageous that Google's model does not require high-end hardware, making it more accessible and cost-effective, at least for this application.

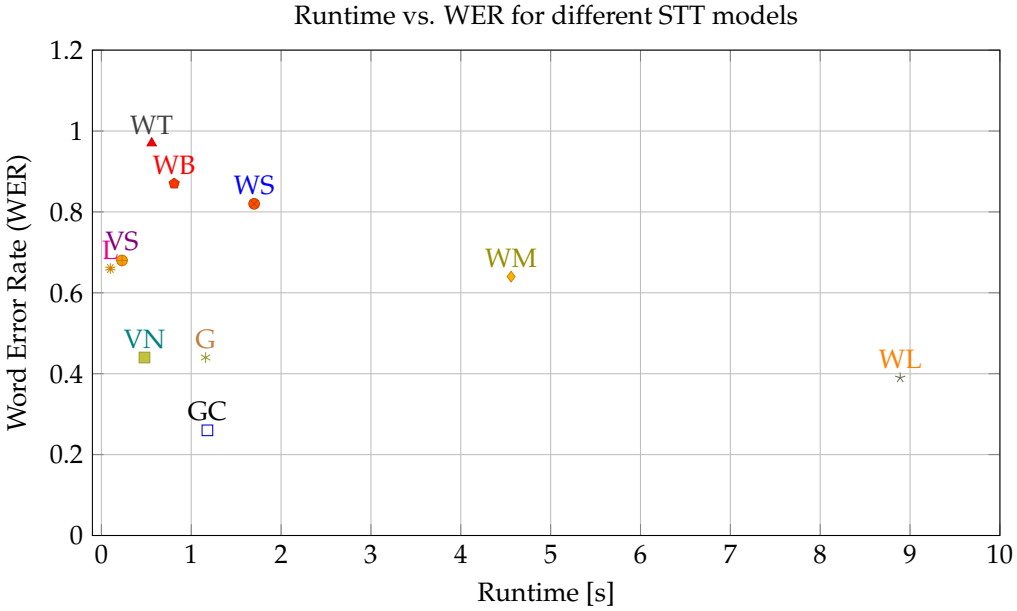


Figure 6. Runtime and WER comparison for various STT models, averaged over 311 instances from the VID1S dataset using an RTX 3080 TI. Model initials near each point are as follows: WB = Whisper Base, WT = Whisper Tiny, WS = Whisper Small, WM = Whisper Medium, WL = Whisper Large, VS = Vosk Small, VN = Vosk Normal, G = Google STT, GC = Google STT using context, L = Leopard.

3.4. Semantic Analysis Results

STT might need to be more robust when dealing with small children, as small mistakes or using different words while retaining the meaning can occur. This could confuse users since the expectation of a voice recognition system in robots, such as the one proposed in this article, is to enable natural interaction. This implies multiple ways of asking the same thing without a fixed set of words. For these reasons, LLMs are used to grasp the intention behind the actions. Due to limited GPU resources for in-line LLMs, we opted to test online services, namely ChatGPT and Gemini from Google. To evaluate the suitability of this tool, we performed an intention analysis on each instruction in the dataset to see if the use of LLMs provides not only actionable results but also serves as a helper to improve understanding of common mistakes made by kids or STT systems.

Table 4 presents the semantic analysis results of the VID audios. From these results, it is evident that Gemini delivered the best performance. Notably, Gemini could comprehend instructions that the STT did not transcribe accurately but still preserved the

meaning, providing a secondary verification extension that enhanced the system's robustness. ChatGPT also demonstrated the ability to interpret instructions correctly, albeit with less accuracy than Gemini. Thus, Gemini was chosen for its superior performance in accurately determining actions and faster runtime.

Table 4. LLM evaluation over Google STT context-aware transcriptions on the VID1S dataset. STT Correct corresponds to the perfectly accurate transcriptions, while the other columns indicate the number of voice instructions that the semantic analysis tools correctly identified along with their runtime. The total number of voice instructions was 311.

	STT Correct	Gemini		GPT-3.5 Turbo		GPT-4	
		Correct	Runtime	Correct	Runtime	Correct	Runtime
Total	184	200	1.2 s	192	1.1 s	186	2.5 s

3.5. System Repeatability Results

We conducted a system repeatability test using VID1S audios that the Google STT service had accurately recognized in previous experiments. To demonstrate the system's repeatability, we designed various tasks requiring the robot to interact with objects within the workspace. We chose tasks such as relocating the red cube to the red bucket and positioning the red star in its designated bucket. Our goal was to assess the system's overall consistency and the reliability of its components, including object detection and robot actions. We compiled audio sequences for each task from VID recordings. We tried to use audio recordings from the same child whenever possible; however, we occasionally had to combine recordings for younger participants. This necessity arose because the system's performance with these participants was limited, making it challenging to generate a coherent sequence of actions from a single four-year-old's recordings. We placed different pieces in the workspace for each task and ran the system using the sequence of recordings, processing them automatically in real time. The results, which we detail in Table 5, unveiled unexpected outcomes. Although we initially had concerns only about robot repeatability and object detection, we discovered the STT system was not as robust as we had thought, which led to some misdetections. Remarkably, we observed no failures in the robot's actions or the vision system.

Table 5. This table provides a breakdown of the total and successful attempts, as well as errors categorized by STT, object detection, and LLM.

Dataset	Total Attempts	Successful Attempts	Errors by STT	Errors by Object Detection	Errors by LLM
VID Tasks	100	93	7	0	0

3.6. Evaluation of the System with Children Aged 4 to 6

We conducted a workshop in a robotics laboratory tailored for children aged 4 to 6, despite our datasets including ages up to 8. This age range was chosen due to the developmental differences between younger children and those older than 6, aiming for a more homogeneous group. The workshop was structured into two one-hour sessions, each accommodating 12 students, effectively covering the same content twice in two hours with a brief transition period. The session included an introduction to robotics, emphasizing our robot's operation, followed by interactive tasks where children commanded the robot to perform simple actions. Figure 7 captures a moment when children interact with the robot during the workshop.

Figure 7. Children engaging with the robot in the workshop session.

The workshop yielded several important observations. Firstly, the activity was highly engaging, with children enthusiastic about interacting with the robot. The students issued 39 voice commands, of which, 15 required repetition due to the system's initial failure to interpret the instructions accurately. Despite these challenges, all participants could successfully command the robot by repeating the command when needed.

Most failures stemmed from the children's inability to formulate instructions within the given time, resulting in incomplete commands. Additionally, developmental differences in speech skills, especially among four-year-olds, presented challenges. This highlights a significant challenge for future improvements: enhancing the system's robustness to accommodate the diverse characteristics of younger children and making the system more adaptable to their unique needs.

These findings are encouraging, demonstrating the system's adaptability and its practicality for use by children in this age range. The option to retry commands significantly boosts the system's utility, ensuring that despite initial miscommunications, children can successfully interact with the robot as intended.

4. Discussion

The development of a voice-controlled robotic system for early education, as presented in this study, marks a step forward in leveraging technology to enhance learning experiences for young children. By employing advanced artificial intelligence technologies, including large language models and sophisticated object detection techniques, the system enables children as young as four to interact with robots intuitively through voice instructions. This approach not only simplifies the interaction process but also opens up a wide range of educational possibilities that were previously inaccessible. The complexities of programming languages and robot theory, often too advanced for young learners to grasp, can no longer be barriers to exploration and learning. Moreover, this type of system enhances accessibility, making educational robotics more inclusive and adaptable to diverse learners' needs.

Our findings demonstrate the system's effectiveness in recognizing and executing commands from children. However, the study also underscores several technological challenges. The voice recognition and action recognition systems need to be significantly faster to prevent distractions among young learners, which could limit their practical use in classrooms. Additionally, these systems must reduce their hardware requirements for broader adoption, making the system more accessible to schools of any size and budget. This represents significant challenges on the technological front to achieve widespread implementation.

Future research should explore more open-ended interaction scenarios, enabling children to perform various tasks and engage more deeply with the robot. Moreover,

comprehensive investigations are necessary from an educational perspective to assess learning gains across multiple domains, such as computational thinking, collaborative work, and subjects like science or mathematics, to substantiate the educational value of this type of robot. Integrating learning subjects could be particularly effective given the robot's versatile nature. This aspect is crucial not only for the educational adoption of the technology but also for mitigating concerns about the use of technology in early education, which is often perceived as more detrimental than beneficial.

Additionally, it is worth noting that the potential of these interfaces and robots extends far beyond education. The key lies in making technology more intuitive to use, fostering true human–robot collaboration, and ensuring its positive application across all areas of human activity.

5. Conclusions

In this work, we demonstrate that children aged 4 to 6 can control collaborative robots using voice commands, unveiling numerous possibilities for educational advancements. Children possess a remarkable capacity for interaction, often constrained by the limitations of current robotics and programming languages unsuitable for their age group. This research shows the feasibility of employing more intelligent robots, enabling children not just to learn robotics but to engage in various tasks, thus broadening the scope of what is achievable in early education through technology and twenty-first-century skills. Additionally, this research introduces the opportunity to explore deeper human–robot collaboration, a vital skill for the future workforce that requires further exploration in early education settings.

Author Contributions: Conceptualization, C.A.A., A.C., C.A. and B.R.; formal analysis, C.A.A., A.C. and B.R.; funding acquisition, A.C. and C.A.; Investigation, C.A.A., A.C., C.A. and B.R.; methodology, C.A.A. and A.C.; resources, A.C.; software, C.A.A.; validation, C.A.A.; writing—original draft, C.A.A.; writing—review and editing, C.A.A., A.C., C.A. and B.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Research and Development Agency through the grants ANID Vinculación Internacional FOVI220011, ANID FONDECYT Iniciación 11220143, ANID FONDEF ID21I10256, and project ID21I10256 Bio-Bio University.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Ethics and Bioethics Committee of Universidad Austral de Chile (22 April 2022).

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

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request. Due to ethical concerns surrounding participant privacy and the specific terms of the informed consent, the data cannot be made publicly available.

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

References

1. Castro, A.; Medina, J.; Aguilera, C.A.; Ramirez, M.; Aguilera, C. Robotics Education in STEM Units: Breaking Down Barriers in Rural Multigrade Schools. *Sensors* **2023**, *23*, 387. [CrossRef] [PubMed]
2. Sisman, B.; Kucuk, S. An Educational Robotics Course: Examination of Educational Potentials and Pre-service Teachers' Experiences. *Int. J. Res. Educ. Sci.* **2019**, *5*, 510–531.
3. Misirli, A.; Komis, V. Robotics and Programming Concepts in Early Childhood Education: A Conceptual Framework for Designing Educational Scenarios. In *Research on e-Learning and ICT in Education: Technological, Pedagogical and Instructional Perspectives*; Karagiannidis, C., Politis, P., Karasavvidis, I., Eds.; Springer: New York, NY, USA, 2014; pp. 99–118. [CrossRef]
4. Garvis, S.; Keane, T. A Literature Review of Educational Robotics and Early Childhood Education. In *Technological Innovations in Education: Applications in Education and Teaching*; Garvis, S., Keane, T., Eds.; Springer Nature: Singapore, 2023; pp. 71–83. [CrossRef]
5. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All You Need. In Proceedings of the Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.

6. Radford, A.; Kim, J.W.; Xu, T.; Brockman, G.; McLeavey, C.; Sutskever, I. Robust Speech Recognition via Large-Scale Weak Supervision. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023.
7. Lv, W.; Xu, S.; Zhao, Y.; Wang, G.; Wei, J.; Cui, C.; Du, Y.; Dang, Q.; Liu, Y. DETRs Beat YOLOs on Real-time Object Detection. *arXiv* **2023**, arXiv:2304.08069.
8. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
9. Cherti, M.; Beaumont, R.; Wightman, R.; Wortsman, M.; Ilharco, G.; Gordon, C.; Schuhmann, C.; Schmidt, L.; Jitsev, J. Reproducible scaling laws for contrastive language-image learning. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 2818–2829.
10. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models From Natural Language Supervision. In Proceedings of the ICML, Virtual Event, 18–24 July 2021.
11. Williams, R.; Park, H.W.; Oh, L.; Breazeal, C. PopBots: Designing an Artificial Intelligence Curriculum for Early Childhood Education. *AAAI Conf. Artif. Intell.* **2019**, *33*, 9729–9736. [CrossRef]
12. Calo Mosquera, N.; García-Rodeja Gayoso, I.; Sesto Varela, V. Construyendo conceptos sobre electricidad en infantil mediante actividades de indagación. *Enseñanza Cienc. Rev. Investig. Exp. Didácticas* **2021**, *39*, 223–240. [CrossRef]
13. Kambouri-Danos, M.; Ravanis, K.; Jameau, A.; Boilevin, J.M. Precursor models and early years science learning: A case study related to the water state changes. *Early Child. Educ. J.* **2019**, *47*, 475–488. [CrossRef]
14. Mendez, E.; Ochoa, O.; Olivera-Guzman, D.; Soto-Herrera, V.H.; Luna-Sánchez, J.A.; Lucas-Dophe, C.; Lugo-del Real, E.; Ayala-Garcia, I.N.; Alvarado Perez, M.; González, A. Integration of Deep Learning and Collaborative Robot for Assembly Tasks. *Appl. Sci.* **2024**, *14*, 839. [CrossRef]
15. Aguilera-Carrasco, C.A.; González-Böhme, L.F.; Valdes, F.; Quitral-Zapata, F.J.; Raducanu, B. A Hand-Drawn Language for Human–Robot Collaboration in Wood Stereotomy. *IEEE Access* **2023**, *11*, 100975–100985. [CrossRef]
16. Leopard-Picovoice Speech-to-Text Engine. Available online: https://picovoice.ai/docs/leopard/ (accessed on 29 January 2024).
17. Vosk Speech Recognition Toolkit: Offline Speech Recognition API for Android, iOS, Raspberry Pi and Servers with Python, Java, C# and Node. Available online: https://github.com/alphacep/vosk-api (accessed on 29 January 2024).
18. Google Cloud Speech-to-Text. Available online: https://cloud.google.com/speech-to-text/ (accessed on 29 January 2024).
19. Team, G.; Anil, R.; Borgeaud, S.; Wu, Y.; Alayrac, J.B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A.M.; Hauth, A.; et al. Gemini: A Family of Highly Capable Multimodal Models. *arXiv* **2023**, arXiv:2312.11805.
20. OpenAI Introducing ChatGPT. OpenAI. 2022. Available online: https://openai.com/blog/chatgpt (accessed on 10 January 2024)
21. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.E.; Fu, C.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
22. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
23. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R.B. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 386–397. [CrossRef] [PubMed]
24. Soviany, P.; Ionescu, R.T. Optimizing the Trade-off between Single-Stage and Two-Stage Object Detectors using Image Difficulty Prediction. *arXiv* **2018**, arXiv:1803.08707.
25. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
26. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; Kwon, Y.; Michael, K.; Fang, J.; Yifu, Z.; Wong, C.; Montes, D.; et al. Ultralytics/yolov5: V7. 0-YOLOv5 SOTA realtime instance segmentation. *Zenodo* **2022**.
27. Tsai, R.; Lenz, R. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Trans. Robot. Autom.* **1989**, *5*, 345–358. [CrossRef]
28. Unity Technologies. Unity Perception Package. 2020. Available online: https://github.com/Unity-Technologies/com.unity.perception (accessed on 9 March 2024).
29. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLOv8. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 9 March 2024).
30. Lin, T.; Maire, M.; Belongie, S.J.; Bourdev, L.D.; Girshick, R.B.; Hays, J.; Perona, P.; Ramanan, D.; Doll'a r, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014.