

**AR EDUCATIONAL SUITE FOR PRIMARY
STUDENTS'
GEOMETRY AND LITERACY**

INDIVIDUAL THESIS

N.N. Wickramasingha

IT22214102

Individual component: Sketch-to-3D Phonics Learning Component

Group ID: R26-IM-005

Dissertation submitted in partial fulfilment of the requirements for the
B.Sc. (Hons) Degree in Information Technology
Specializing in Interactive Media

Department of Information Technology
Sri Lanka Institute of Information Technology

Sri Lanka

October 2026

DECLARATION AND SUPERVISOR CERTIFICATION

Candidate declaration. I declare that this thesis is my own work and does not incorporate, without acknowledgement, material previously submitted for a degree or diploma at any university or institute of higher learning. To the best of my knowledge, it contains no material previously published or written by another person except where acknowledgement is made in the text. I grant the Sri Lanka Institute of Information Technology the non-exclusive right to reproduce and distribute this thesis, in whole or in part, in print, electronic, or other media, while retaining the right to use the content in future work.

Academic integrity and AI disclosure. I have disclosed all use of generative AI and other automated tools in accordance with the module policy. I remain responsible for the accuracy, originality, citations, code, data, analyses, and conclusions in this thesis.

Student name: N.N.Wickramasingha

Student ID: IT22214102

Signature: _____ Date: _____

Supervisor certification. The above candidate carried out this research for the bachelor's degree thesis under my supervision.

Role and name	Signature	Date
Supervisor: Mr. Aruna Ishara Gamage		
Co-supervisor: Mr. Nushkan Nizmi		

ABSTRACT

Early vocabulary learning requires children to associate objects, written words and spoken language, yet many interactive applications isolate these practices or begin with predefined content. This individual study designs and evaluates Component C4 of the GeoLex AR Learning Suite: a sketch-driven, multisensory vocabulary activity for primary learners. Children draw one of five supported objects (apple, cat, fish, sun or tree) on a Unity canvas; a convolutional neural network trained using 25,000 Quick, Draw!-derived examples is executed locally through Unity Sentis. Recognition selects a predefined 3D model, whose relative size reflects the original sketch bounds. Correct spelling and offline Vosk target-word recognition independently advance the model from low to medium to full opacity, after which Android augmented-reality placement is unlocked. The held-out QuickDraw test accuracy was approximately 96.4%, which is not a measure of performance on children's drawings. The group study reports 25 C4 sessions with children aged 6–8: spelling completion in 20/25 (80%), target-word speech acceptance in 21/25 (84%), and complete spelling–speech–AR progression in 17/25 (68%). Mean session duration was 56.52 seconds; aggregate pre-test and post-test means increased from 4/10 to 8/10. Because child-sketches intended-object ground truth and participant-level paired learning scores were not retained, neither child-sketch recognition accuracy nor statistical significance is claimed. The principal individual contribution is a reproducible, on-device sketch-to-model-to-vocabulary learning workflow with progressive task-gated AR reinforcement, including its data logging and interface design. The study supports feasibility while identifying reliability, pronunciation validity and extended child testing as priorities.

Keywords: Augmented Reality, Sketch Classification, Vocabulary Learning, Convolutional Neural Network, Multisensory Interaction

ACKNOWLEDGEMENTS

I express my appreciation to Mr. Aruna Ishara Gamage, my supervisor, and Mr. Nushkan Nisme, my co-supervisor, for their academic guidance, design feedback and encouragement throughout the research and development process. I also acknowledge the Department of Information Technology and the Interactive Media academic staff at the Sri Lanka Institute of Information Technology for the technical and research environment that supported this work.

I thank my fellow researchers in group R26-IM-005 for developing the complementary geometry and literacy components, discussing system integration and contributing to the evaluation of the overall educational suite. I am grateful to the participating children, their parents or guardians, and supporting educators for enabling the learner-centred study described in the group research manuscript. The assistance of family and friends during development and documentation is also appreciated.

TABLE OF CONTENTS

1 INTRODUCTION.....	1
1.1 Overall project context	1
1.2 Individual component context	2
1.3 Problem statement	2
1.4 Research gap.....	3
1.5 Aim and objectives	5
1.6 Research questions	6
1.7 Scope, limitations and assumptions	7
1.8 Individual contribution	7
1.9 Thesis structure.....	8
2 LITERATURE REVIEW AND RELATED WORK	9
2.1 Review method and scope	9
2.2 Domain and theoretical foundations.....	9
2.3 Sketch recognition and visual learning.....	10
2.4 AR vocabulary learning and multimodal reinforcement	10
2.5 Air-writing, speech recognition and responsibility boundaries	11
2.6 Mobile AI, on-device inference and AR frameworks.....	11
2.7 Research gap synthesis and conceptual framework.....	11
3 RESEARCH METHODOLOGY	13
3.1 Research design.....	13
3.2 Dataset sources and sampling.....	14
3.3 CNN model design and experimental procedure	15
3.4 Child pilot sample and procedure	15
3.5 Evaluation metrics and operational definitions	16
3.6 Data analysis and inference boundaries.....	16
3.7 Ethics, privacy and responsible AI	17
3.8 Reproducibility and research artifacts	17
4 PRODUCT AND SOLUTION DESIGN	18
4.1 Functional and non-functional requirements	18
4.2 C4 architecture within the complete solution	18
4.3 Design alternatives and final decisions.....	19
4.4 Sketch canvas and preprocessing design	20
4.5 Sketch-size transfer and prefab normalization.....	20
4.6 Progressive model reveal and learning-state invariants.....	21
4.7 Spelling and speech task design	22
4.8 AR placement, manipulation and exit flow	22
4.9 Child-centred interface and accessibility considerations	22
5 IMPLEMENTATION INTEGRATION AND TESTING	23
5.1 Development environment and technology stack	23
5.2 Sketch capture, session start and recognition inference.....	23
5.3 Model selection, progress controllers and prefabs.....	23
5.4 Learning-flow and task uniqueness implementation	24
5.5 Offline Vosk lifecycle and failure management	24
5.6 AR scene and gesture implementation	25
5.7 User-interface assets, naming and package safety.....	25
5.8 ResearchDataLogger structure and data extraction	26
5.9 Integration strategy with C1, C2 and C3	26
5.10 Software verification and testing plan	27
5.11 Deployment and operational boundaries	28
6 RESULTS AND VALIDATION	29
6.1 Overview of available evidence	29
6.2 CNN evaluation on held-out QuickDraw images	29
6.3 C4 workflow results from children aged 6–8	29
6.4 Timing, attempts and confidence.....	31

6.5 Learning assessment outcomes.....	31
6.6 Functional and system observations	32
6.7 Integration results and evidence status	32
6.8 Summary of findings against research questions.....	32
7 DISCUSSION	33
7.1 Interpretation of technical and educational findings.....	33
7.2 Comparison with literature and closest work	33
7.3 Research contribution and novelty	33
7.4 Product and engineering contribution.....	34
7.5 Limitations and threats to validity	34
7.6 Implications for child-centred design and evaluation.....	34
8 COMMERCIALISATION AND SUSTAINABILITY.....	35
8.1 Value proposition and target users	35
8.2 Market and competitive positioning	35
8.3 Pricing, costs and sustainability.....	35
8.4 Intellectual property, privacy and compliance.....	36
8.5 Scalability and contribution to SDG 4.....	36
9 CONCLUSION AND FUTURE DIRECTIONS	37
9.1 Conclusion.....	37
9.2 Achievement of individual objectives	37
9.3 Individual research and product contributions.....	38
9.4 Future directions.....	38
REFERENCES.....	39
GLOSSARY.....	41
APPENDICES	42
Appendix A - Research log data dictionary.....	42
Appendix B - Suggested learner observation and test instruments.....	42
Appendix C - Example acceptance test procedures.....	43
Appendix D - Academic integrity and AI-use disclosure.....	44

LIST OF FIGURES

Figure 3.1: Equal class distribution in the documented CNN training, validation and test partitions ...	15
Figure 4.1: Logical C4 validation-state transitions for independent spelling and speech tasks	19
Figure 4.2: Three-stage visibility progression and conditional AR access.....	21
Figure 6.1: Observed C4 completion rates in the 25 child-user sessions; derived from the group pilot	30
Figure 6.2: Descriptive C4 pre-test and post-test group means, each on a 10-mark scale	31

LIST OF TABLES

Table 1.1: Individual research objectives and verifiable outputs	6
Table 2.1: Research families and the C4 design position.....	12
Table 3.1: QuickDraw sampling plan used in C4 CNN development.....	14
Table 3.2: C4 outcome definitions, data sources and valid interpretations	16
Table 4.1: Requirement traceability from child action to system response	18
Table 4.2: Proposal-to-implementation changes relevant to research claims.....	20
Table 5.1: Verification matrix for final device and group integration.....	28
Table 6.1: C4 child-user pilot outcome summary (n = 25)	30
Table 8.1: Exploratory commercial and cost assumptions from the original proposal.....	36
Table 9.1: Concluding objective assessment with unambiguous evidence labels	38
Table A.1: C4 anonymised session fields and interpretation	42
Table B.1: Candidate short child observation items (not retrospectively claimed)	43

LIST OF ABBREVIATIONS

AI	- Artificial Intelligence
APK	- Android Package Kit
AR	- Augmented Reality
ARCore	- Google augmented reality platform for Android
CNN	- Convolutional Neural Network
CSV	- Comma-Separated Values
DSR	- Design Science Research
FBX	- Filmbox 3D exchange format
JSON	- JavaScript Object Notation
LTS	- Long-Term Support
ML	- Machine Learning
NFR	- Non-Functional Requirement
ONNX	- Open Neural Network Exchange
PNG	- Portable Network Graphics
RQ	- Research Question
SDG	- Sustainable Development Goal
UI	- User Interface
URP	- Universal Render Pipeline
Vosk	- Offline speech recognition toolkit
XR	- Extended Reality

1 INTRODUCTION

1.1 Overall project context

The group implementation contains four distinct research components. C1 addresses mathematical validation of learner-drawn geometric nets. C2 provides fault-specific correction and virtual slicing of solids. C3 follows camera-observed fingertip trajectories to recognize air-written English letters. C4, the subject of this thesis, begins with a child-produced object sketch and develops vocabulary learning through recognition, spelling, speech and AR exploration. The geometric pathway was evaluated with children aged 9–12, whereas the literacy pathway used children aged 6–8; the two age bands reflect different educational requirements [1]. Shared application-level navigation and reinforcement do not imply that the four components use the same classifier, sensor or evaluation measure.

The educational motivation for the suite is grounded in the distinction between passively viewing a digital representation and actively producing an artefact that the system must interpret. In conventional learning software, a pupil may select an already labelled object and receive a corresponding animation. Here, the learning sequence starts with a net, a movement, or a sketch created by the learner. This changes the instructional role of computation: the application must decide what the input represents, whether it is acceptable for the task, and what feedback should follow. The same broad sequence can support different curriculum domains without suggesting that a single recognition model is suitable for all four components.

Within the geometry pathway, the instructional challenge is to connect two-dimensional representations to three-dimensional spatial relationships. C1 tests whether a proposed net is mathematically compatible with a supported solid, while C2 supports correction and virtual slicing. Within the literacy pathway, the emphasis moves to visible symbols, familiar objects, word formation, and spoken language. C3 processes the temporal movement of a fingertip in free space; C4 processes the visual appearance of an object sketch and uses its predicted class as the vocabulary target. This distinction is important because success in geometric validation cannot be interpreted as success in language recognition, and vice versa.

1.2 Individual component context

The individual component considered in this dissertation is C4, initially proposed as Multisensory Phonics Integration and subsequently developed as the Sketch-to-3D Phonics Learning Component in the GeoLex AR Learning Suite. Its instructional function is best described as sketch-driven vocabulary learning. The learner produces a drawing, receives a recognized object category, practices the corresponding English spelling, attempts the target spoken word and explores a matching 3D object in augmented reality. Although the product label retains the word “phonics”, this implementation does not teach or measure systematic grapheme–phoneme decoding. That conceptual distinction is maintained throughout the methods and findings [1].

The learner uses a touch-enabled white drawing canvas rather than capturing a paper sketch through the camera. The application preprocesses the resulting bitmap for a five-class convolutional neural network covering apple, cat, fish, sun and tree. A class prediction is mapped to a curated Unity prefab; the object is not a newly reconstructed mesh. In addition, the original drawing bounds are measured independently from the normalized classifier input. This supports a relative display-size relationship, giving the pupil a visible connection between the extent of the drawing and the ensuing model without claiming dimensional reconstruction.

Once the model is selected, two independently assessed activities determine educational progression. The spelling task accepts the exact target word after whitespace trimming and case normalization. The speech task uses offline Vosk recognition to determine whether the target word was recognized, rather than grading phonemes or accent. Each distinct completed activity increases model opacity, and only completion of both conditions enables horizontal-plane AR placement. The learner can rotate and resize the placed object and return to the drawing screen to begin another session. Research logging captures task events, durations, acceptance and model confidence for later descriptive analysis. These mechanisms define the boundary of C4, separately from the group’s camera-based air-writing component C3.

1.3 Problem statement

This study asks how a constrained, mobile and locally executed learning component can turn a freehand child sketch into a repeatable sequence of automated recognition,

accessible vocabulary tasks and AR reward, while preserving a meaningful separation between implementation success and educational outcome. The implementation must operate within ordinary Android device resources, support children who may read only short instructions, and gather non-identifying interaction evidence that can be interpreted without overstating classifier or speech accuracy. The component is intentionally limited to five recognizable concepts so that the whole process can be developed and assessed in a final-year research setting.

The engineering problem includes several interdependent requirements. The drawing interface must be sufficiently forgiving for developing motor control while still yielding a bitmap suitable for recognition. Preprocessing must prepare the bitmap for the model without discarding all useful information about the original drawing. The prediction must be translated into a fixed word and model reference, and the system must prevent progression when an educational task has not been completed. These dependencies make the problem more substantial than displaying a classifier label on screen.

The instructional problem is equally important. Young learners may interpret a visually appealing 3D reward as the objective and skip the language tasks if the application allows it. The system therefore needs explicit state transitions, clear feedback and a rule that links reward access to task completion. Such gating is a design decision about educational progression; it is not evidence that the reward automatically produces better vocabulary retention.

1.4 Research gap

Sketch-to-model research, including view-aware reconstruction, demonstrates that computer-generated three-dimensional representations can be derived from drawing inputs, and sketch CNN research demonstrates recognition of object categories [2]–[4]. Those technical achievements do not in themselves prescribe a child-friendly

educational workflow. Conversely, AR vocabulary research reports benefits from visual interactivity and form–meaning association but often initiates learning with predetermined vocabulary content [5]–[7]. The intersection of learner-generated object input with task-by-task vocabulary reinforcement is therefore the design space investigated here.

The narrower gap is an integrated process with explicit stage dependencies. The sketch must be recognized before its word can be established; spelling and speech must be independently accepted before the AR reward becomes available; visible model progress must reflect completed tasks rather than arbitrary button presses. The group research paper frames the overall contribution in this way: four component-specific analyses implement a common input–analysis–feedback–reinforcement contract [1]. For C4, the research contribution is not a new foundational CNN architecture, a generative 3D system, or a phoneme-scoring algorithm. It is a reproducible assembly and evaluation of interacting mechanisms applied to learner-originated vocabulary content.

Existing sketch recognition work provides methods for assigning object categories to hand-drawn images, and sketch-to-model research investigates methods of recovering or generating 3D form [2]–[4]. These strands show the feasibility of interpreting sketches, but the research question addressed here is not whether arbitrary 3D geometry can be reconstructed. C4 deliberately uses predefined assets. The gap concerns the design of an instructional sequence that starts with a learner-originated sketch and turns a category prediction into age-appropriate language practice.

AR-supported vocabulary studies emphasize motivation, visual context and links between word forms and meanings [5]–[7]. Yet the presence of AR alone does not establish how a child’s own sketch should select the word, how incomplete learning tasks should be represented, or how access to reinforcement should be controlled. C4 joins these concerns through explicit task flags and a staged representation of the recognized object. This is a workflow and interaction contribution, not a claim to have invented CNN classification, speech recognition or AR plane detection.

1.5 Aim and objectives

The main aim is to design, implement and evaluate an Android-based sketch-driven multisensory vocabulary learning component that interprets a learner’s own digital sketch, visualizes a corresponding size-responsive 3D object, validates spelling and spoken target-word recognition, and unlocks interactive AR reinforcement only after both educational tasks have been completed. The design follows an iterative Design Science Research (DSR) approach, in which a functioning artifact is built, examined and refined using technical and learner-centred evidence [8], [9]. Table 1.1 summarises the operational objectives.

The objectives are deliberately arranged as a dependency chain. Recognition (O1) establishes a bounded semantic label; visualization (O2) maps that label to a prepared model while retaining a relative sketch-size cue; vocabulary interaction (O3) supplies independently checked written and spoken activities; AR gating (O4) ties the reward to those activities; and evaluation (O5) tests the complete system using technical and child-session evidence. Failure at an earlier stage can affect later outcomes, which is why full-session completion is a useful workflow measure but not a substitute for measuring each stage.

O1 can be examined with held-out labelled image data and reproducible model preprocessing. O2 can be verified by inspecting the class-to-prefab mapping and comparing model scale under different occupied drawing bounds. O3 requires separate tests for accepted and rejected spelling input and spoken target-word outcomes. O4 requires both positive and negative state-transition tests, including an attempt to enter AR after only one completed task. O5 combines development evaluation with observational records from the pilot rather than asserting causal educational effectiveness.

ID	Objective	Documented evidence or acceptance condition
O1	Recognize a learner sketch	Five-class CNN; saved sketch; class label and confidence
O2	Relate sketch to a 3D visualization	Predefined prefab selected; original sketch-size ratio maps to relative model scale
O3	Validate two vocabulary activities	Exact spelling match; offline Vosk target-word acceptance; distinct task flags
O4	Gate an interactive AR reward	AR enabled only when both tasks completed; horizontal placement; rotate and resize

O5	Evaluate operation and usability	JSON/CSV logs; held-out classification measure; 25-session child workflow study
----	----------------------------------	---

Table 1.1: Individual research objectives and verifiable outputs

1.6 Research questions

RQ1 asks whether a compact CNN trained on five categories can provide a viable recognition stage for user-generated digital sketches, and what can legitimately be inferred from held-out benchmark evaluation. RQ2 asks how the original sketch's spatial extent can be preserved as a relative parameter when the classification image is normalized. RQ3 asks whether an explicit two-task progression mechanism can reliably connect spelling and speech recognition to 3D visibility and AR unlocking. RQ4 asks how often children in the available pilot reach each learning state and complete the full AR sequence. RQ5 asks what the observed pre-/post-test comparison suggests about learning and what methodological limitations prevent stronger causal conclusions. These questions keep recognition accuracy, workflow completion and educational change distinct [1].

These research questions are complementary rather than interchangeable. RQ1 addresses the feasibility and limits of image classification. RQ2 addresses the design choice to retain spatial information that would otherwise be lost during resizing. RQ3 examines correctness of the learning-state logic and its connection to visual progression. RQ4 focuses on what actually happened during child use. RQ5 considers observed learning change while recognizing that a pre-/post-test difference alone cannot establish causation.

The questions also guide interpretation of negative outcomes. An unsuccessful session might reflect an unrecognized sketch, a spelling error, a rejected spoken attempt, an interaction difficulty or an AR issue. Without stage-specific evidence it would be inappropriate to assign all unsuccessful sessions to the classifier. Accordingly, the available logs and group results are discussed as workflow evidence, while missing measurements are identified as limitations.

1.7 Scope, limitations and assumptions

The application recognises only apple, cat, fish, sun and tree from touchscreen sketches. Runtime AI inference uses Sentis, while Vosk checks target spoken words. Spelling is an exact normalized text comparison; five predefined 3D objects have low, medium and full visibility, followed by horizontal-plane AR placement. The original FastAPI server was a historical prototype, not the current inference deployment.

The main exclusions are unrestricted recognition, bespoke geometric reconstruction, stroke-order assessment, phoneme-level pronunciation scoring and causal claims about learning. The 96.4% classifier result is from held-out QuickDraw examples, not independently labelled child sketches. Because normal child logs do not record the intended object, they cannot supply child-sketch accuracy. The short pilot and absence of paired assessment records further limit generalisation [1].

Operational boundaries were chosen to keep the artifact testable. The current workflow assumes a touchscreen drawing surface, five supported classes, a compatible Android device, and access to the relevant local inference and AR capabilities. A suitable lighting environment and a detectable horizontal surface can affect AR placement, while speech acceptance may be affected by background noise, microphone quality and a child's speaking style. These are practical operating conditions, not participant characteristics inferred from the pilot.

The component does not provide unconstrained semantic understanding. A sketch of an unsupported item must still be interpreted by a five-class classifier unless additional rejection logic is implemented and validated. Nor does size-responsive prefab selection mean that the system estimates real-world object dimensions. The mapping preserves a relative aspect of the drawing activity for visualization; it does not establish a metric reconstruction of the object.

1.8 Individual contribution

My C4 responsibility covers the drawing interface, local sketch classification, original-sketch size mapping, 3D model states, spelling, offline target-word recognition, AR placement and interaction logging. Other group members are responsible for C1 geometric-net validation, C2 correction and slicing, and C3 air-writing trajectory analysis. This dissertation does not attribute their independent algorithms to C4.

The individual contribution is situated within a shared group system. The C4-specific work begins with designing a child-facing drawing surface and preparing sketch images for a bounded recognition task. It includes connecting a recognized category to a vocabulary word and prepared 3D prefab, preserving drawing bounds for relative scale, implementing separate spelling and spoken-word acceptance states, and controlling access to AR placement. Interaction records then support analysis of how learners progress through those stages.

Integration responsibilities should be described accurately. Shared navigation, Unity runtime conventions and AR presentation connect the component to the broader suite, but the algorithms for geometric-net validity, correction and slicing, and air-writing trajectories belong to the other members. Where a shared asset or service is reused, the thesis should describe the interface and dependency without claiming sole authorship of group-wide infrastructure.

1.9 Thesis structure

Chapters 2–9 cover literature, methodology, design, implementation, results, discussion, sustainability and conclusions.

2 LITERATURE REVIEW AND RELATED WORK

2.1 Review method and scope

The literature review is a focused, critical synthesis rather than a claim of a systematic meta-analysis. It uses the sources selected for the individual proposal, the group's final research paper and relevant primary technical documentation. The main themes are learner-generated input, sketch classification, AR-supported word learning, on-device execution and spoken-word processing. Relevance to the implemented five-class C4 interaction was preferred over merely listing all possible AR and AI methods. The distinction between conceptual support and direct empirical comparison is maintained throughout: a paper demonstrating 3D reconstruction, for example, cannot be treated as an accuracy benchmark for this application's five-class classification task.

2.2 Domain and theoretical foundations

AR places a virtual object in a real environment, allowing a learner to inspect it from different positions rather than only looking at a flat card. Reviews of AR in K–12 education document promising educational effects but also stress usability and context-dependent variation [10], [11]. For C4, AR is a conditional reward after written and spoken practice. Placing the digital model at the end avoids allowing attractive visualization to bypass the assessed vocabulary tasks. This sequencing gives a concrete instructional purpose to the technology while acknowledging that the reported study did not experimentally isolate AR's independent effect.

The current application title refers to phonics, but the learning activity is operationalized here as early vocabulary and spelling association. The system recognizes whole spoken words and exact written labels; it does not explicitly teach letter–sound correspondences or analyse phonemes. This distinction aligns with the group research paper's C4 description, which deliberately avoids a formal phonics claim [1]. It also makes the outcome measures defensible: spelling completion and target-word speech acceptance are observable program states, whereas pronunciation quality or long-term decoding ability would require different instruments.

2.3 Sketch recognition and visual learning

Research on sketch understanding spans retrieval, classification and geometric reconstruction. Sketch2Model illustrates how a view-aware method can infer a 3D representation from a single drawing [2]. Recognition-oriented studies use visual features to identify the category of a sketch [3], while representation-learning work explores how sketch and handwriting modalities can be encoded [4]. These are related but not interchangeable problems. Reconstruction attempts to infer geometry; classification outputs a semantic label. C4 uses the second task because its educational purpose is to establish the correct target word from a limited familiar-object vocabulary.

Sketch inputs vary in line thickness, continuity, size, stroke count and position. A child's drawing may omit parts that adults treat as diagnostic or may add features absent from benchmark examples. For that reason, a preprocessing pipeline should be evaluated as part of the model rather than as a visually neutral operation. Normalizing an image to a fixed 64×64 grayscale representation assists the classifier, but it can also remove information about the original dimensions of the marks. C4 deliberately separates the recognition representation from the sketch-size variable so the task can retain one feature of the learner's input while standardizing what is presented to the CNN.

2.4 AR vocabulary learning and multimodal reinforcement

Korosidou reports AR-related motivation and language-learning outcomes among young learners [5]. Belda-Medina and Marrahi-Gomez discuss vocabulary acquisition and student motivation with AR [6]. Lin and Wei review vocabulary learning in AR/VR gamification contexts and identify interactivity and vocabulary mapping as recurring design concerns [7]. These sources support the educational relevance of connecting words to interactive representations, but their different contexts, participants and instructional treatments do not supply a directly transferable effect size for C4. The present implementation must therefore be evaluated on its own workflows and learners.

2.5 Air-writing, speech recognition and responsibility boundaries

Speech-to-text is also different from phonetic scoring. The offline Vosk toolkit returns recognition hypotheses for spoken audio using a local recognizer [12]. C4 accepts a pronunciation task when the expected object name is recognized. Such acceptance is evidence that the recognizer returned a matching word under the current conditions; it does not prove that the child articulated each sound with a validated threshold of phonetic quality. Accent variation, noise, age and recognizer vocabulary influence performance. A future phonics assessment would require a speech corpus with expert labels, an explicit scoring criterion and reliability testing against human judgements. The present thesis treats Vosk only as a target-word interaction mechanism.

2.6 Mobile AI, on-device inference and AR frameworks

The design's computational foundation separates training from inference. TensorFlow/Keras is used to train the image model, while Unity Sentis imports the trained model for local prediction during application use [13]. The official Sentis documentation explicitly defines it as an inference library capable of executing neural networks using CPU or GPU resources [13]. This arrangement eliminates a runtime dependence on a self-hosted Python prediction endpoint. A development-time server may still have supported earlier experimentation, but that historical architecture is not the final implemented learning path. The difference affects reproducibility, offline access, privacy and system diagrams.

2.7 Research gap synthesis and conceptual framework

Table 2.1 contrasts C4 with three relevant technology families. The gap is not an absolute claim that no app has ever included drawing, speech and 3D. Instead, the literature and project rationale support a specific design question: how can a learner-created sketch determine the vocabulary target, and how can two distinct assessed tasks control visible 3D progression before AR access? This focus reduces the tendency to equate feature count with research novelty and clarifies the internal validity of the artifact evaluation. The working hypothesis is architectural: interpretable state transitions make the educational experience testable and loggable.

Table 2.1: Research families and the C4 design position

Approach	Typical strength	Limit relative to this study
Sketch recognition / reconstruction	Interprets drawings or infers 3D shape	Does not necessarily link recognition to spelling, speech and learning-state gating
AR vocabulary applications	Visual form–meaning feedback	Frequently start from predetermined illustrations/word sets
Speech or air-writing tools	Specialised word or movement input analysis	Usually lack learner-sketch reference and shared 3D reinforcement
C4 developed workflow	Child-originated sketch drives five-class word target	Limited class vocabulary; predefined model; no validated phoneme scoring

3 RESEARCH METHODOLOGY

3.1 Research design

Two evidence streams are considered. The technical stream describes the CNN dataset and held-out model evaluation, the execution path in Unity and the intended functionality of the principal state transitions. The learner stream uses the C4 subset of the group pilot involving 25 children aged 6–8 and the reported pre-test/post-test means [1]. This deliberate separation prevents an appealing accuracy figure from a standardized training dataset being misread as a result on real child sketches. It also prevents task completion rates from being interpreted as class-specific prediction accuracy or causal learning improvement. The observational pilot supports feasibility and identifies problems; it is not a randomized controlled trial.

The methodological unit of analysis differs between the two evidence streams. In the technical stream, a labelled image is the unit used to evaluate the five-class classifier. In the child-user stream, one application session is the unit for describing whether the learner progressed through the spelling, speech and AR states. Treating these as distinct units is necessary because the 3,750 held-out images and the 25 child sessions do not represent interchangeable samples. The first concerns model discrimination under a controlled dataset distribution; the second concerns whether an educational sequence can be completed in an authentic interaction setting.

The design-science perspective treats the operational software component as the research artifact. Its relevance is established through the learning task: an object selected and drawn by a child is interpreted and used as the basis of subsequent activities. Its design is expressed through separable mechanisms for image preprocessing, recognition, model selection, task verification and AR unlocking. Evaluation therefore examines both artifact behavior and the limited learner evidence reported in the group study. The research does not include a control condition, random assignment or delayed post-test, and no causal effect should be inferred from the observed pre/post means.

An important methodological choice is to use explicit transition criteria instead of treating a displayed 3D model as evidence that learning has occurred. The recognized category supplies a candidate vocabulary item. The spelling task and speech-

acceptance task each produce a distinct completion state. The AR reward is conditional on their conjunction. This architecture allows the pilot log to distinguish partial completion from full completion, making the observed 68% full-flow completion meaningful as a workflow measure rather than as an image-recognition score.

3.2 Dataset sources and sampling

The CNN development set was assembled from bitmap images derived from Google’s Quick, Draw! dataset, a large publicly available collection of labelled drawings [14]. The documented C4 model uses exactly five categories: apple, cat, fish, sun and tree. Each category contributes 5,000 examples, giving 25,000 bitmap samples in total. Per class, 3,500 samples are allocated to training, 750 to validation and 750 to held-out testing. The totals are therefore 17,500, 3,750 and 3,750 respectively. The same allocation for each class avoids creating a trivial majority-class advantage in overall classification accuracy. Table 3.1 gives the five balanced data partitions and Figure 3.1 visualises their equal allocation.

The held-out test partition evaluates generalization to unseen QuickDraw examples drawn from the same source distribution. That is a useful model-development checkpoint but does not answer whether a particular child’s submitted drawing is correctly recognised. The final application uses a finger-drawn Texture2D image that may differ in aspect, canvas margin, line weight and drawing behaviour. For child-sketch classification accuracy, a study would first need the child to identify the intended class independently of the model prediction or use an external ground-truth annotation. The normal C4 free-draw logger does not include that field, so its mean prediction confidence cannot fill this gap.

Class	Training	Validation	Held-out testing	Total
Apple	3,500	750	750	5,000
Cat	3,500	750	750	5,000
Fish	3,500	750	750	5,000
Sun	3,500	750	750	5,000
Tree	3,500	750	750	5,000
TOTAL	17,500	3,750	3,750	25,000

Table 3.1: QuickDraw sampling plan used in C4 CNN development

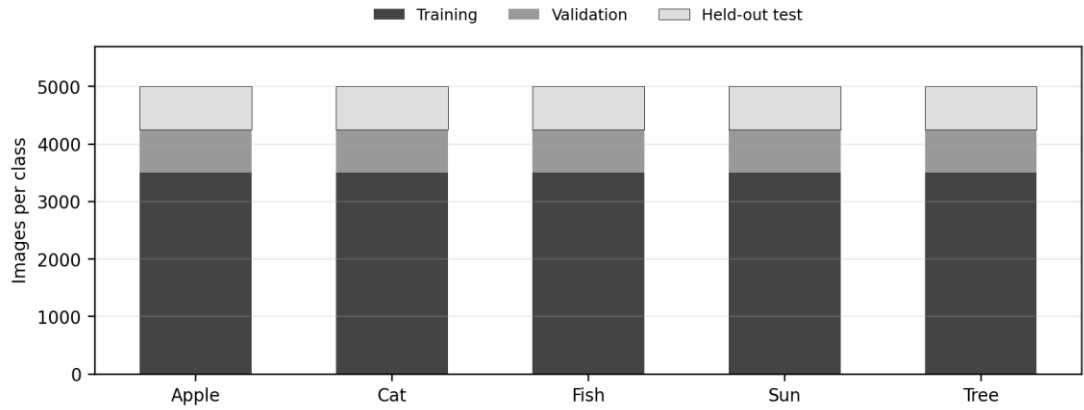


Figure 3.1: Equal class distribution in the documented CNN training, validation and test partitions

3.3 CNN model design and experimental procedure

The CNN is trained in TensorFlow/Keras using 64×64 grayscale inputs rescaled by a factor of $1/255$. Its approximate architecture contains three convolutional stages with 32, 64 and 128 filters, each followed by max pooling, then dropout, flattening and a 128-neuron rectified linear hidden layer. The final dense layer outputs five softmax scores. The training configuration uses Adam optimization, sparse categorical cross-entropy loss, batch size 64 and up to 15 epochs. Early stopping monitors validation loss with patience of three epochs, limiting continued training when validation performance no longer improves. This architecture is a deliberately constrained classifier rather than a large general vision model.

The documented supervised workflow begins with labelled class examples and a fixed index-to-label mapping, followed by training/validation separation and a held-out final assessment. A model exported to Unity must preserve class ordering, expected input channels and numerical scaling. A class-index mismatch is especially consequential: a technically valid softmax tensor can silently activate the wrong vocabulary label and 3D prefab. Reproducibility therefore depends on packaging the class mapping alongside the trained model, checking the input/output tensor conventions, and comparing a set of known test examples between TensorFlow and the Unity runtime.

3.4 Child pilot sample and procedure

The group pilot recruited 50 different learners across two age bands under the institutional ethics process described in the research paper [1]. Twenty-five learners

aged 9–12 interacted with geometry components C1 and C2. A separate cohort of 25 learners aged 6–8 used literacy components C3 and C4. The latter cohort is relevant here. The same 25 children were counted for both literacy components; thus the C4 sample is 25, not an additional 25 unique participants beyond the group total. This distinction matters for accurate sample-size reporting and any interpretation of the joint educational study.

3.5 Evaluation metrics and operational definitions

Five outcome categories are distinguished. First, held-out dataset accuracy describes CNN classification on labelled QuickDraw testing images. Second, spelling completion proportion is the number of child sessions in which exact object-name spelling was eventually accepted divided by all C4 sessions. Third, speech acceptance proportion is the number of sessions in which the Vosk target-word condition became true divided by all sessions. Fourth, full workflow completion is the fraction of sessions in which both task flags were true and AR placement was completed. Fifth, learning change is the reported difference between the mean post-test and pre-test marks. The measured times and attempts describe operational friction rather than correctness. Table 3.2 defines each evaluation measure and its valid meaning.

Measure	Evidence source	What the measure can establish
Held-out CNN accuracy	Labelled QuickDraw test set	Accuracy only for held-out QuickDraw distribution
Mean predicted confidence	25 child-user logs	Average model score; not child-sketch accuracy
Spelling completed	Task-state log	Whether exact expected word was accepted
Speech accepted	Offline Vosk task state	Whether target spoken word was recognised
Both tasks + AR	Task flags and AR events	Completion of complete designed interaction
Pre/post average	Group research manuscript	Descriptive mean score change; no inferential claim

Table 3.2: C4 outcome definitions, data sources and valid interpretations

3.6 Data analysis and inference boundaries

Descriptive statistics were drawn from the group research paper’s reported C4 results. Completion rates are computed with 25 sessions as the denominator; $20/25 = 80\%$, $21/25 = 84\%$, and $17/25 = 68\%$. Because the model required both unique tasks to unlock AR, the 17 successful full-flow sessions correspond to those in which both

conditions were satisfied. The arithmetic also permits a limited descriptive overlap: three sessions completed spelling without speech acceptance, four obtained speech acceptance without spelling completion, and one satisfied neither. These are derived counts, not a new independent dataset. No class-level recognition result is inferred from them.

3.7 Ethics, privacy and responsible AI

The evaluated users were children, so research participation requires additional safeguards. The group paper states that institutional ethics procedures were followed, parents or guardians gave informed consent, learners gave assent, and records were anonymized [1]. C4’s current logger uses pseudonymous identifiers such as P001_S001, rather than names, and stores task data in the application’s private persistent directory. Such identifiers reduce direct identification but do not make all metadata harmless: timestamps, sketches and audio can still create privacy risk if exported carelessly. Consent and approved retention rules must therefore govern any collection or sharing beyond the published aggregate evidence.

3.8 Reproducibility and research artifacts

A reproducible technical account records the Unity editor version (2022.3.62f3 LTS), the five supported class labels, the exact 25,000-image data partition, CNN preprocessing and training configuration, the use of Sentis inference, the 3D prefab mappings and the fields written by ResearchDataLogger. These details are essential because changes to image resizing, the order of class indices or normalized model scale can silently change predictions or visual output. A deployable archive also needs package-lock information, Unity project settings, scene files, relevant .meta GUIDs, model assets and an unambiguous Android package identifier.

4 PRODUCT AND SOLUTION DESIGN

4.1 Functional and non-functional requirements

The functional scope starts with safe capture of a drawing: the system must accept freehand touch strokes in a bounded white area, distinguish an empty canvas from a valid sketch, support clearing, and save a PNG copy. The next requirement is prediction of one of five recognized labels through an on-device model. An accepted label must select a corresponding 3D preview, while the occupied drawing bounds should influence the prefab's relative displayed scale. The validation screen must accept and separately track spelling and speech tasks, provide understandable success or retry feedback, advance object visibility after each distinct task and permit AR access only after both distinct tasks are completed. Table 4.1 traces user actions to software acceptance conditions.

ID	Requirement	Acceptable visible/system outcome
FR-01	Input and reset sketch	Touch marks recorded; Clear empties pad; blank Continue rejected
FR-02	Recognise five categories offline	Label and confidence returned by Sentis without runtime server
FR-03	Visualise selected model with size relation	Correct predefined model; relative scale derived from bounds
FR-04	Validate spelling	Trimmed case-insensitive exact comparison; retry on mismatch
FR-05	Accept target speech	Vosk recognition matches word or offers retry
FR-06	Gated reveal and AR	0/1/2 distinct tasks maps to low/medium/full; AR only at 2
FR-07	AR interaction and navigation	Horizontal placement; drag rotate; pinch resize; Back resets to sketch
FR-08	Research data capture	Session JSON/CSV records task attempts, states and times

Table 4.1: Requirement traceability from child action to system response

4.2 C4 architecture within the complete solution

The current architecture separates Unity screen handling, drawing logic, inference, learning-task control, preview presentation, AR placement and research logging. At the centre is LearningFlowManager, which maintains the current recognized label, task-state flags, feedback and progression. SketchDrawingPad owns the texture and sketch bounds. SketchInputManager validates the input, initiates local inference and

starts the study session. SentisInferenceManager owns the CNN runtime interface. ModelDisplayManager selects and scales the proper low-, medium- or full-opacity prefab. OfflinePronunciationManager coordinates Vosk listening, and ARPlacedModelManager handles placement and user manipulation. UIScreenManager coordinates the intro, sketch and validation screens. Figure 4.1 illustrates the two independent learning-task branches.

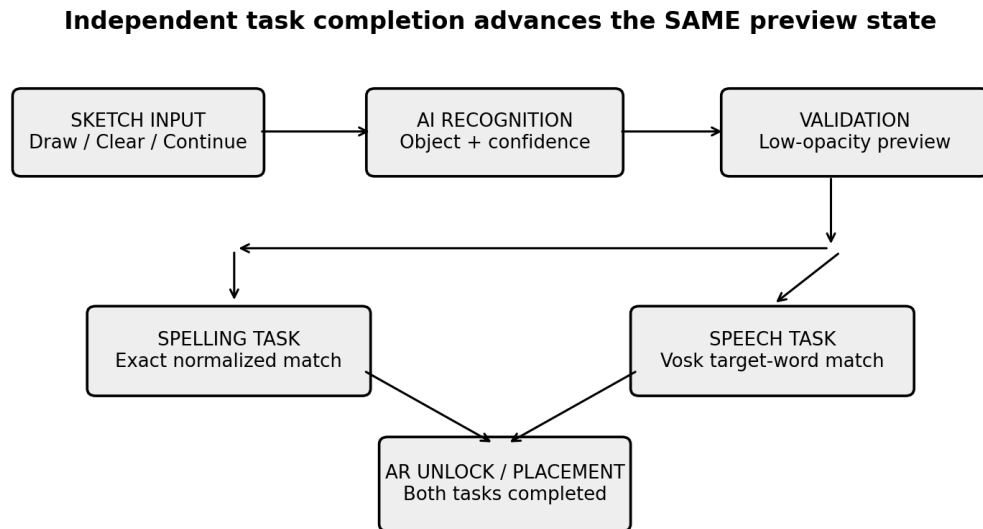


Figure 4.1: Logical C4 validation-state transitions for independent spelling and speech tasks

4.3 Design alternatives and final decisions

The original proposal envisaged camera capture of physical sketches, actual geometric sketch-to-3D generation and air-written word matching. The final learning flow favours a bounded interactive drawing pad, semantic classification to existing objects and typed whole-word spelling. These are substantive design changes, not small presentation changes. Digital sketches provide a controlled blank background and immediate access to the texture used for classification; predefined models ensure consistent educational labels and suitable artwork for a limited vocabulary; typed spelling isolates word recall from the challenging technical task of camera-based air-writing. Component C3 separately investigates trajectory-based writing, so this division also avoids attributing the same recognition work to two students. Table 4.2 specifies the proposal-to-final implementation differences.

Initial proposal expectation	Current C4 implementation	How reported in this thesis
Camera-based sketch recognition	Finger drawing into Unity Texture2D	Digital freehand input; no paper-camera evaluation
Custom sketch-generated 3D geometry	Predefined 3D class prefabs	Semantic classification plus relative scale mapping
Air-written word input from gesture module	Typed exact word matching	Spelling validation, not air-writing accuracy
Pronunciation quality validation	Offline Vosk whole-word hypothesis matching	Target-word speech acceptance only
Server/TensorFlow Lite considered	TensorFlow training, Sentis local inference	On-device deployment; historical FastAPI prototype
Broad 6–12 target ages	C4 pilot cohort aged 6–8	Age-specific evaluation, n = 25

Table 4.2: Proposal-to-implementation changes relevant to research claims

4.4 Sketch canvas and preprocessing design

SketchDrawingPad captures strokes into a white Texture2D, with a black brush and a typical 512×512 drawing texture. ClearDrawing restores the blank surface; HasDrawing signals whether meaningful strokes exist; SaveSketchAsPng persists an image for development or research review. Decorative UI background art includes the frame around the white area, but the actual interactive DrawingArea object must remain above that image so it can receive touch events. Decorative backgrounds therefore disable Raycast Target, while touch-sensitive elements remain enabled. This small layering decision is essential: replacing the functional drawing pad with a static illustration would preserve appearance but destroy the data pipeline.

4.5 Sketch-size transfer and prefab normalization

Let w_s and h_s be the occupied horizontal and vertical stroke extents and let W and H be the unnormalized drawing texture dimensions. A normalized extent ratio is $r = \max(w_s/W, h_s/H)$, clamped to a valid interval. The preview-scale mapping is then $s = [s_{\min} + (s_{\max} - s_{\min})r] \times m$, where $s_{\min}$ and $s_{\max}$ are configured prefab-scale bounds and m is a multiplier. The documented illustrative values are $s_{\min}$ approximately 0.3, $s_{\max}$ approximately 1.2 and m approximately 1. These are project configuration examples, not calibrated real-world dimensions; AR also applies its own smaller scale multiplier.

4.6 Progressive model reveal and learning-state invariants

Each recognized object has three visual prefab stages: low-opacity, medium-opacity and full-opacity. The intended presentation is approximately 25% opacity immediately after classification, approximately 60% after one unique task and 100% after both unique tasks. Two Boolean completion states, `spellingCompleted` and `pronunciationCompleted`, determine the stage. With neither flag set, the model remains low-opacity. With exactly one flag set, it becomes medium-opacity. With both set, the full model appears and View in AR becomes available. The same learner can perform the tasks in either order without changing the final condition. Figure 4.2 summarises this staged reveal.

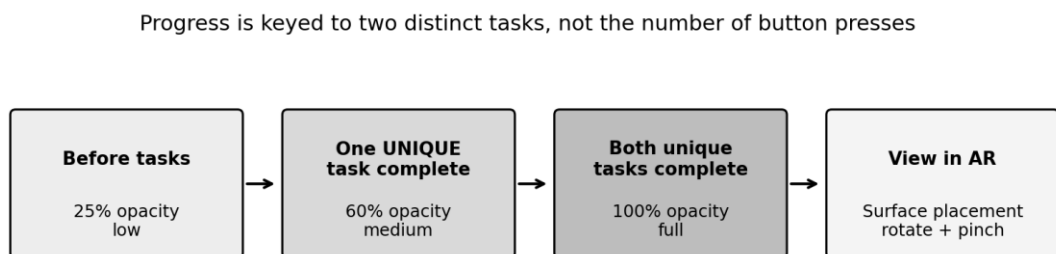


Figure 4.2: Three-stage visibility progression and conditional AR access

The underlying logic can be stated as an invariant: the reveal stage equals the number of distinct completed tasks, capped at two. This means that two successful spelling submissions do not substitute for one successful spelling submission plus one accepted spoken-word attempt. Similarly, returning to a previously opened panel must not reset a true completion flag or increment progress a second time. Negative tests are as important as positive ones: the AR button should remain locked when neither flag is set and also when only one is set. The two-task approach provides an observable educational dependency, but it should not be interpreted as an experimental demonstration that opacity changes improve learning.

4.7 Spelling and speech task design

The spelling target is exactly the label returned by the sketch classifier, for example apple. Before comparison the input is trimmed and converted to a common case, so APPLE, Apple and apple are accepted, while aple, apples and appel are rejected. A mismatch displays a brief child-friendly retry message and increments the number of attempts; an exact match marks the spelling task complete once. This decision is intentionally transparent and reproducible. It tests recall of a whole word in a five-word vocabulary. It does not attempt to score letter formation, detect near-misses as partly correct or assess mastery of a broader spelling system.

In the speech task, a child taps Speak, grants microphone permission when necessary and says the current target word. OfflinePronunciationManager starts or reuses an initialized Vosk recognizer, parses the transcription result and accepts a target-word match. The manager must guard against overlapping attempts and should not initialize the speech engine anew after every press. A nonmatching hypothesis prompts retry, with failures and recognized text logged as available. Speech completion advances the same two-task state. These operations support a pronunciation-oriented activity, but the underlying acceptance mechanism remains word recognition; no human-rated articulation score is produced.

4.8 AR placement, manipulation and exit flow

AR availability is gated on both task flags, not simply on reaching a particular screen. The Android implementation uses AR Foundation and ARCore with an AR Session, XR Origin, AR camera, plane tracking and raycast-based touch placement [15]. When the learner selects View in AR, the camera view is activated and the application identifies suitable stable horizontal-up planes. A valid touch places the corresponding fully revealed object on a detected surface with a small vertical offset. Plane visuals can then be hidden to reduce visual clutter, while tracking remains active as needed. User interaction consists of one-finger rotation and two-finger pinch resize.

4.9 Child-centred interface and accessibility considerations

The portrait interface uses illustrated controls, brief visual tutorials and touch-blocking overlays while preserving the functional drawing pad and Unity callbacks.

5 IMPLEMENTATION INTEGRATION AND TESTING

5.1 Development environment and technology stack

The documented standalone C4 scene is `C4_Main_AR_Learning_Scene`, created in Unity 2022.3.62f3 LTS with the Universal Render Pipeline for an Android APK in portrait orientation. Core logic is written in C#. The CNN was trained using TensorFlow/Keras and exported for Unity-compatible local inference. Unity Sentis performs runtime prediction, while Vosk and its Unity wrapper provide offline speech recognition [13], [12]. AR Foundation, the ARCore XR Plugin and XR Plugin Management provide the Android real-world placement functions [15]. Blender and imported FBX prefabs supply the visual vocabulary objects. TextMesh Pro supplies UI text and entry fields.

5.2 Sketch capture, session start and recognition inference

The Unity drawing path begins with `SketchDrawingPad`, which exposes `DrawingTexture`, `HasDrawing`, `ClearDrawing()`, `GetSketchSizeRatio()` and `SaveSketchAsPng()`. On Continue, `SketchInputManager` checks that strokes exist and that local inference has been initialized. It temporarily disables repeated Continue submissions, saves the image, obtains the occupied-size ratio and starts a research session. It then supplies the texture to `SentisInferenceManager` and receives a class label with its confidence. The result is logged before `LearningFlowManager.BeginLearning()` opens the validation sequence. If the classifier returns no usable label or is unavailable, the user should see a retry message and the button should be restored, rather than an empty or incorrect model.

5.3 Model selection, progress controllers and prefabs

The `ModelDisplayManager` maps each category to a group of prepared prefabs. Naming follows a component-specific convention, for example `C4_PF_Apple_LowOpacity`, `C4_PF_Apple_MediumOpacity` and `C4_PF_Apple_Full`. Equivalent sets exist for Cat, Fish, Sun and Tree. A preview point located relative to the ordinary Unity camera holds the current model. When the recognised label is set, the manager destroys the previous preview, chooses the proper

initial variant and applies the sketch-derived scale. LearningFlowManager subsequently selects the middle or full variant according to the two independent task flags. This split separates rendering details from educational decision logic.

5.4 Learning-flow and task uniqueness implementation

LearningFlowManager acts as the authoritative state owner for the current learning episode. BeginLearning(label, ratio) establishes the target, clears previous task values and creates the initial model. SelectSpellingTask opens the input panel; CheckSpelling compares the normalized entry and logs attempts. SelectPronunciationTask opens the speech panel; the actual speech timer should start when listening begins, not when the panel is merely displayed. CompletePronunciationTask flags acceptance while PronunciationFailed displays a retry instruction. UpdateLearningProgress counts unique completed modalities and instructs ModelDisplayManager to select the appropriate visual state. ViewInAR remains inactive until both flags are true.

5.5 Offline Vosk lifecycle and failure management

OfflinePronunciationManager encapsulates microphone permission, Vosk initialization, recording, recognition callbacks and result parsing. The documented wrapper exposes IsInitialized, IsInitializing and IsRecording, together with StartVoskStt, StartRecording and StopRecording capabilities. These states prevent duplicate initialization and overlapping active sessions. A first attempt may trigger resource initialization; later attempts should reuse the existing recognition engine. Reusing the recognizer matters particularly on mobile hardware, where loading model assets and constructing audio processing objects for each attempt can introduce unnecessary delay and memory instability.

A realistic test matrix includes permission granted, permission denied, empty speech input, a nonmatching recognized word, noisy input, multiple Speak presses and navigation away during recording. The component should never show Pronunciation accepted solely because the Speak button was touched; acceptance requires the expected word in a parsed result. Nor should it preserve a partial audio task into the next child's session. Since no separate labelled speech corpus or human-rated articulation scores were supplied, this implementation description is not a claim of

word-error rate, dialect fairness or valid pronunciation-quality scoring. It is a precise statement of what the software attempts to recognize.

5.6 AR scene and gesture implementation

ARPlacedModelManager enables the AR view after successful learning progression. The project uses AR Session and XR Origin with an AR camera; an AR raycast identifies appropriate upward horizontal planes. The placed prefab uses a small global AR scale multiplier (approximately 0.10) and a small surface offset (approximately 0.002) in documented typical settings. Rotation sensitivity was approximately 0.20, and pinch behavior uses configurable speed and scale limits. These values describe the intended project setup and remain subject to practical device calibration. Because a child's drag may begin on a UI button, UI-hit filtering is needed to prevent unintended object rotation during navigation.

5.7 User-interface assets, naming and package safety

The intended project-specific asset root is Assets/C4_SketchTo3DPhonics, with child folders for AI assets, models, materials, prefabs, scenes, scripts and UI. C4_ prefixes on GameObject names distinguish objects in the merged Unity hierarchy. The renaming operation should be completed in the Unity Project window to preserve .meta GUIDs. Project-specific UI sprite replacements can be made without deleting working buttons or input components, which keeps serialized Unity Inspector assignments intact. Full-screen decorative images use Raycast Target disabled; transparent actual buttons retain hit testing and OnClick callbacks.

The special-case folders are the imported dependencies. StreamingAssets, Vosk model folders, XR, TextMesh Pro and the existing Assets/Assets speech package structure should not be moved blindly. Although renaming a GameObject often preserves direct serialized references, scripts that call `GameObject.Find("OldName")` or `transform.Find("OldName")` may break because those strings do not follow Unity rename operations. Likewise, a MonoBehaviour script should not be renamed inconsistently with its public class. A safe merge begins with a backup, proceeds through limited reference-preserving changes and ends with targeted smoke tests after each batch rather than a wholesale refactor.

5.8 ResearchDataLogger structure and data extraction

ResearchDataLogger creates pseudonymous per-session records such as P001_S001.json and an aggregate C4_FreeDraw_UserTesting.csv file under Application.persistentDataPath/ResearchLogs. The field set includes ParticipantID, SessionID, start/end timestamps, predicted object, AI confidence, sketch-size ratio, spelling attempts and completion, speech attempts and acceptance, AR opened/placed and session duration. Multiple drawings by the same learner increment the session suffix rather than creating a new participant ID. The logger is intended to detect the highest existing session number after application restart so that files are not overwritten. It records the learning process, not an independent clinical or educational diagnosis.

On Android, app-private storage is not necessarily visible through a normal file browser. A developer can inspect it through Android Debug Bridge, sometimes by adb run-as in a debuggable development build, or through an accessible Android/data path depending on operating-system restrictions. The actual package name must be read from the Unity Android Player settings before forming such commands. Exported records should be checked for the correct columns, pseudonymous identifiers, monotonically increasing sessions and complete end states. Importantly, the default schema lacks an ExpectedObject or AI prediction correctness field; analysts cannot manufacture child-sketch accuracy simply from PredictedObject and AIChecked.

5.9 Integration strategy with C1, C2 and C3

The four-member system is integrated through a common application shell and navigation model while retaining component-local learning logic. C1's mathematical validation and C2's geometry correction/slicing are closely coupled; C3's air-writing trajectory model is separate from C4's digital-sketch word association. The current group paper explicitly states that C3 and C4 share some literacy-level navigation, visual assets and AR presentation but use separate learner inputs [1]. Thus C4's present typed spelling cannot be relabelled as C3 camera recognition merely because both capabilities exist somewhere in the overall application.

A conservative integration sequence is to preserve C4's standalone scene and prefabs, adopt consistent C4 naming, inspect every Inspector serialization link, connect shared

entry and exit navigation, and then verify the child journey in the merged Unity project. Dependencies such as AR Session, XR Origin and EventSystem should be deduplicated only after checking ownership. Test cases should verify that returning from C4 does not activate old cameras or leave speech recordings active, and that re-entering C4 resets task flags. The source materials identify integration readiness checks as outstanding after component organization; therefore this chapter does not claim that every merged-project regression test has already passed.

5.10 Software verification and testing plan

Verification is organized at unit, integration and Android acceptance levels. Unit-level or deterministic checks should focus on blank-sketch rejection, spelling normalization, task uniqueness and scale mapping. Integration checks should cover Unity texture-to-Sentis mapping, callback handling, prefab selection, Vosk permission/lifecycle management, C4 navigation and logging. Android checks should exercise AR plane detection, placement, one-finger rotation, pinch scaling, microphone permission and persistence across relaunch. The supplied development record describes these features as implemented or substantially implemented, but it does not attach a complete executed test report for all cases. For academic integrity the matrix below is labelled a verification plan rather than a fabricated historical pass/fail log. Table 5.1 provides the prospective verification matrix.

Test	Action or input	Expected check	Evidence status
T01	Press Continue on blank canvas	Rejected with child-friendly instruction	To verify in final build

T02	Submit each of five object drawings	Matching label + appropriate model	Model mappings documented
T03	Enter APPLE then aple	Exact normalized match / retry	Validation logic documented
T04	Complete spelling twice	Model must remain middle stage	Uniqueness rule documented
T05	Accept both tasks in either order	Full model + AR unlock	17 pilot flows completed
T06	Tap AR surface and manipulate	Place, rotate and pinch	Pilot includes AR placement
T07	Back from AR; draw again	New clean session and canvas	To regress after merge
T08	Relaunch app and export logs	Session numbering and CSV safe	To verify on device
T09	Disable microphone permission	Recoverable spoken-task error	To verify on device
T10	Rename hierarchy/relocate assets	All Inspector/asset paths preserved	To regress after merge

Table 5.1: Verification matrix for final device and group integration

Verification should distinguish deterministic assertions from tests that depend on uncertain device conditions. Spelling comparisons, unique-task counters and size interpolation can be checked with fixed input–output cases. Local speech recognition depends on microphone permission, acoustic conditions and model initialization. AR plane detection depends on device support, lighting and trackable surfaces. A defensible final test record would contain the executed device configuration, observed output and evidence for each case; merely listing the expected outcomes in a test plan does not demonstrate that the group-integrated APK has passed them.

5.11 Deployment and operational boundaries

The final application is intended to run as an Android APK on ARCore-compatible hardware in portrait orientation. The classifier and Vosk operate offline at runtime, while camera-based AR requires environmental permissions and tracking availability. Users need a sufficiently bright, textured flat surface for stable plane detection, and speech should be attempted in a reasonably quiet setting. A standard phone may support sketch and vocabulary activities yet still fail AR eligibility; that limitation should be communicated clearly rather than hiding the View in AR button without explanation. Device compatibility screening belongs to deployment planning.

6 RESULTS AND VALIDATION

6.1 Overview of available evidence

The results draw from two explicitly different evidentiary sources: the documented TensorFlow/Keras held-out experiment and the C4 data summarized in the group’s final ICAC research manuscript [1]. The model experiment uses 3,750 previously unseen labelled QuickDraw images and provides a benchmark classification accuracy. The learner pilot uses 25 actual child-user C4 sessions and provides completion states, attempts, model confidence and duration. The pre-test/post-test result is available only as aggregate group means for the C4 cohort. The supplied records do not provide complete case-level responses, split-by-object errors or device-by-device AR reliability. The following presentation does not invent those missing observations.

6.2 CNN evaluation on held-out QuickDraw images

The documented test accuracy is approximately 96.4% on the 3,750-image held-out QuickDraw subset. The original recorded floating-point output was 0.9639999866485596, and this is appropriately rounded to one decimal place in percentage form. The balanced five-class distribution reduces the risk that overall accuracy merely reflects overrepresentation of a single category. Nevertheless, all five classes were sampled from the same corpus and processed under one experimental pipeline. It follows that the result validates in-distribution classification, not deployment accuracy for the touchscreen sketches created by children aged 6–8.

6.3 C4 workflow results from children aged 6–8

The final group study reports 25 C4 learner sessions. Spelling was completed in 20 sessions, a rate of 80.0%. Pronunciation-oriented target-word acceptance occurred in 21, or 84.0%. Both task conditions were satisfied in 17, or 68.0%, and exactly those 17 sessions unlocked and placed the AR model [1]. The task gating therefore appears to have enforced the intended dependency for the observed sessions: the complete AR outcome did not exceed the joint spelling-and-speech completion count. This is a meaningful workflow result, not proof that the CNN predicted the intended object in

68% of sketches. Table 6.1 presents the pilot outcomes and Figure 6.1 compares completion rates.

The difference between individual task rates and full-flow rate is informative. If the model unlocked AR after either task, a much larger fraction of users might have seen the reward. Requiring both tasks means a failure or incomplete attempt in either modality blocks progression. Using only the reported counts, 17 sessions completed both tasks, three completed spelling alone, four had speech acceptance alone and one completed neither. This pattern suggests that an interface offering recovery, guidance and retry is important to reach the final educational state. However, without case-level logs it is not possible to determine why particular children did not finish, whether the cause was model recognition, interface confusion, noise or time constraints.

Table 6.1: C4 child-user pilot outcome summary (n = 25)

Observed measure	Reported result	Interpretation
Spelling completed	20/25; 80.0%	Whole-word exact-match task reached accepted state
Target speech accepted	21/25; 84.0%	Vosk returned the expected object name
Both completed and AR placed	17/25; 68.0%	Full designed vocabulary-to-AR sequence completed
Mean predicted model confidence	0.832; SD 0.099	Confidence across logged predictions; not accuracy
Mean spelling attempts	1.40	Average number of spelling tries
Mean pronunciation attempts	1.28	Average number of target-speech tries
Mean total session time	56.52 s; SD 13.22 s	Overall elapsed C4 session duration

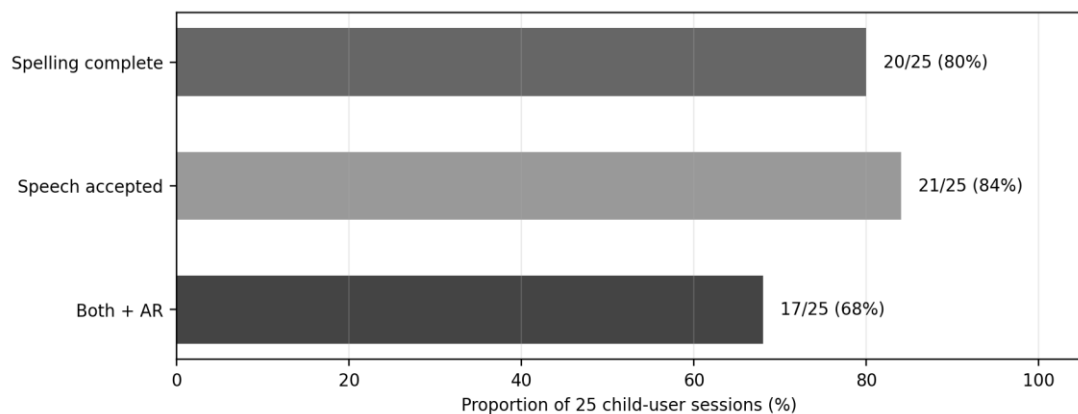


Figure 6.1: Observed C4 completion rates in the 25 child-user sessions; derived from the group pilot

6.4 Timing, attempts and confidence

The mean total session time was 56.52 seconds with a standard deviation of 13.22 seconds. This includes the entire C4 progression as recorded by the application and should not be presented as the average CNN inference time or the time required for speech decoding alone. The mean number of spelling attempts was 1.40 and the mean number of pronunciation-oriented attempts was 1.28 [1]. These averages suggest that some sessions required retry, but they do not identify how often retry was due to a child's misunderstanding versus a technical error. Detailed event timestamps would enable stage-specific analysis and stronger design recommendations in later evaluation.

6.5 Learning assessment outcomes

The C4 cohort's reported pre-test average was 4 marks out of 10 and post-test average was 8 marks out of 10, a difference of four marks or 40 percentage points relative to the ten-mark scale [1]. This is the aggregate descriptive learning result available for C4. It indicates that the observed group performed better on the later assessment. It does not reveal individual gains, the proportion of children who improved or the variance of gains. A four-mark difference also cannot be identified as the effect of AR alone, because the whole experience included drawing, recognition, spelling and speech, and a separate comparison group was not used. Figure 6.2 displays these two aggregate assessment means.

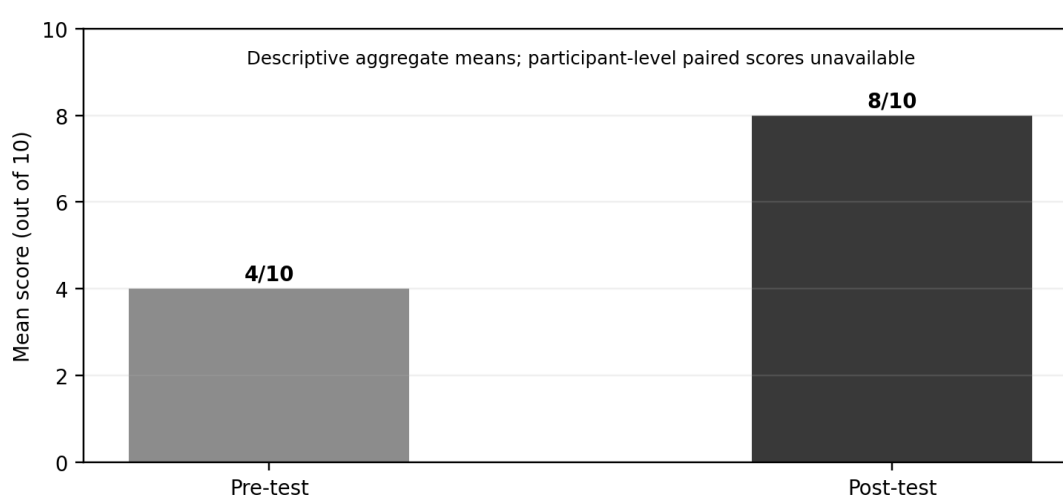


Figure 6.2: Descriptive C4 pre-test and post-test group means, each on a 10-mark scale

A four-mark gain on a ten-mark scale is a descriptive difference between two group means. It should not be relabelled as a 100% relative improvement in literacy, a standardized effect size or statistically significant evidence. The underlying paired scores would be needed to calculate the distribution of gains, uncertainty intervals and within-participant comparisons. Without a control group, repeated-exposure effects and ordinary short-term practice remain plausible explanations alongside the intervention. The reported result is still useful for pilot feasibility, because it indicates that later scores were higher in the observed cohort and motivates a more controlled follow-up study.

6.6 Functional and system observations

Seventeen of the twenty-five child sessions completed the C4 AR journey [1]. The current development record additionally identifies implemented drawing, local inference, spelling, Vosk speech acceptance, progress reveal and touch AR manipulation. However, the provided files do not include a comprehensive post-merge Android regression report, so completed workflow observations should not be mistaken for proof that every integrated deployment case passes.

6.7 Integration results and evidence status

The group manuscript presents a common input–analysis–feedback architecture across four components and 50 learners divided between geometry and literacy cohorts [1]. C4 contributes its own independent sketch-and-vocabulary pipeline within this shared presentation. The manuscript supports the overall design and C4 pilot, but does not supply a complete final post-merge Inspector-reference or APK regression audit.

6.8 Summary of findings against research questions

Overall, C4 demonstrates a feasible task-gated AR workflow, while child-sketch accuracy, phonetic quality and learning causality remain unverified.

7 DISCUSSION

7.1 Interpretation of technical and educational findings

The technical and educational results should be interpreted as complementary rather than interchangeable. The QuickDraw accuracy supports feasibility of a five-class visual recognizer under labelled source-distribution conditions. The child pilot supports feasibility of a connected sequence that uses object predictions as learning targets and requires two tasks to unlock AR. The learning-score comparison documents better aggregate performance after interaction. None of these three results alone proves that the application correctly understands most child-drawn objects or causes durable vocabulary gains. Their combination instead shows how a working artifact may be evaluated at several levels without collapsing distinct research questions into one headline number.

7.2 Comparison with literature and closest work

The work is conceptually aligned with AR vocabulary research showing the value of interactivity and form–meaning association [5]–[7]. It differs in placing the learner’s own sketch at the beginning of the activity rather than treating the word-object pair as fully prescribed. The appropriate original contribution is therefore the carefully managed learning-state sequence. It is not defensible to claim that the limited pilot outperforms the cited AR systems, because those studies differ in learning content, intervention duration, population, devices and performance instruments. A fair future comparison would use the same age group, lesson objectives and assessments for a C4 sketch-originated condition and a predefined-object condition.

7.3 Research contribution and novelty

The individual research contribution is a structured and loggable learner-generated-input pipeline for a restricted vocabulary task. The classifier establishes a semantic anchor; the original sketch bounds supply a separate personalized display parameter; spelling and speech establish two independently completed interaction conditions; and the same object passes through visibility stages before AR exploration. This interaction contract is the source of C4’s novelty within the wider suite. It is made reproducible

through explicit controller responsibilities, task flags and research event fields rather than by claiming an unreported new neural-network algorithm.

7.4 Product and engineering contribution

The engineering contribution is the coordinated Unity application architecture. It joins Texture2D drawing, local CNN inference, class-to-prefab mapping, independent lesson-state tracking, offline speech recognition, conditional AR placement and structured session logging. Component-local scripts and C4-prefixed objects make the resulting project easier to hand over to a group that also maintains separate geometry and air-writing components. The low-/medium-/full-opacity variants represent a simple deterministic visual-progress language that can be inspected in a test rather than relying entirely on animation timing. The return flow supports repeated sketches by the same child under separate session IDs.

7.5 Limitations and threats to validity

The most substantial measurement limitation is missing intended-object ground truth for each child drawing. A predicted class and confidence value cannot establish whether the system correctly interpreted the child's idea. The second limitation is the speech construct: target-word recognition does not measure phoneme-level accuracy, articulation or general English pronunciation skill. The third is that the learning test is reported only through aggregate pre-/post-test means, preventing participant-level gain distributions, paired tests and analysis of attrition. The fourth is the sample size: 25 child C4 sessions provide useful descriptive observations but limited generality across schools, ages, devices or linguistic backgrounds [1].

7.6 Implications for child-centred design and evaluation

For educators, C4 shows how learner drawing can initiate linked word practice. For developers, local inference requires careful model packaging and mobile permission management. For researchers, separate outcome definitions matter: an accurate benchmark classifier need not be accurate on child drawings, and an accepted target-word hypothesis is not a phonetic pronunciation score. Future studies should gather independently labelled sketches, stage-level timing and paired educational marks.

8 COMMERCIALISATION AND SUSTAINABILITY

8.1 Value proposition and target users

The proposed product is a mobile educational activity that begins with a child-generated sketch and supports object–word association through spelling, speech and AR interaction. Its main users are early-primary learners, with teachers and parents acting as facilitators. A classroom teacher may use the activity as a supervised small-group station rather than a substitute for structured reading instruction. A parent may use it for short home-based vocabulary practice where connectivity is unreliable. Institutions might value a low-hardware-dependency tool that does not require specialist sensors beyond an AR-compatible Android device. These are plausible market uses; the research pilot does not measure buying intent or establish national demand.

8.2 Market and competitive positioning

The individual proposal considered primary schools in Sri Lanka, EdTech companies, parents and home-learning users as possible markets [16]. It also proposed annual school licensing, mobile subscriptions and optional premium content. These are exploratory commercial hypotheses, not revenue already earned. C4 would compete with inexpensive or free vocabulary flashcards, spelling exercises, interactive literacy games and AR learning demonstrations. Compared with a conventional spelling app, C4 adds learner-produced sketch recognition and an AR reward; compared with a high-fidelity sketch-to-3D generator, it offers constrained educational content and predictable model presentation. The distinction should be communicated without claiming market-leading recognition or proven learning superiority.

8.3 Pricing, costs and sustainability

The initial proposal suggests illustrative commercial pricing: annual school licences from LKR 30,000–50,000 for small schools, LKR 50,000–100,000 for medium schools and LKR 100,000 or more for large schools; a home subscription of LKR 600–900 monthly or LKR 5,000–8,000 annually; and optional content packs at approximately LKR 1,500–3,000 [16]. These are planning assumptions recorded

before the final component scope was constrained to five items. They must not be presented as validated willingness-to-pay or an approved pricing schedule. A pricing experiment should follow demonstrations with local schools and parents. Table 8.1 records the indicative commercial assumptions.

Area	Proposal estimate / strategy	Evidence status
Small school licence	LKR 30,000–50,000 per year	Suggested, not market-tested
Medium / large licences	LKR 50,000–100,000 / 100,000+ annually	Suggested, not market-tested
Family subscription	LKR 600–900 monthly; 5,000–8,000 annually	Suggested, not market-tested
Optional learning pack	LKR 1,500–3,000	Suggested, not market-tested
Development budget	LKR 155,000 planned total	Proposal estimate, not audited expenditure

Table 8.1: Exploratory commercial and cost assumptions from the original proposal

8.4 Intellectual property, privacy and compliance

Commercial release requires a licence audit for QuickDraw-derived content, Vosk models, Unity assets and third-party plugins. Original project code and graphics may be protected by copyright, but external assets are governed by their own licences [14]. School use must minimise child data, restrict exports and record consent and retention policies. Offline processing is useful but cannot replace privacy and security governance.

8.5 Scalability and contribution to SDG 4

The five-class prototype could grow through validated vocabulary categories, curated 3D assets and teacher-facing functions. Expansion requires additional model testing, content licensing, device QA and consideration of varying reading levels. The research aligns in purpose with Sustainable Development Goal 4 [17], but a limited ARCore-compatible pilot does not demonstrate universal access or national learning impact. Inclusive non-AR modes and curriculum support should be prioritised in a commercial version.

9 CONCLUSION AND FUTURE DIRECTIONS

9.1 Conclusion

This individual research has developed and documented C4, a sketch-driven multisensory vocabulary component within the GeoLex AR Learning Suite. The implemented workflow accepts a learner's touchscreen sketch, uses a locally executed CNN to recognize one of five supported object categories, maps the label to a prepared 3D prefab with sketch-relative scale, and guides the learner through exact spelling and offline spoken target-word activities. Each unique completed task reveals a more visible model stage; both are required to unlock AR placement and manipulation. This design demonstrates a coherent input–analysis–task–reinforcement architecture for an age-appropriate mobile learning activity.

The evidence is promising but bounded. The CNN reached approximately 96.4% accuracy on labelled held-out QuickDraw images. In the available pilot with 25 children aged 6–8, 80% completed spelling, 84% obtained target-word speech acceptance and 68% completed the full sequence and placed the AR object. The reported assessment mean increased from 4/10 to 8/10. These results support technical and instructional feasibility, not generative sketch reconstruction, child-sketch recognition accuracy, phoneme-level pronunciation scoring or proven causal learning improvement. Maintaining these distinctions is central to the credibility of the conclusion.

9.2 Achievement of individual objectives

Objective O1 is supported by a five-class trained CNN and a documented on-device inference pipeline; its benchmark performance has been measured, but recognition accuracy on independently labelled child sketches remains unmeasured. O2 is represented by semantic prefab mapping and original-sketch-relative scale, not custom shape reconstruction. O3 is realised through exact spelling comparison and Vosk target-word acceptance, with separate state flags and retry. O4 is represented by staged 3D reveal and conditional AR horizontal placement, with seventeen complete pilot journeys observed. O5 is addressed through JSON/CSV logging and the descriptive pilot results, while the absence of detailed paired learning records limits stronger

claims. Table 9.1 differentiates the documented outcomes from further verification needs.

Objective	Status described by supplied records	Still required for stronger claim
O1: Classifier	Implemented; 96.4% held-out QuickDraw evaluation	Independent labelled child-sketch test
O2: 3D correspondence	Implemented prefab selection and relative sizing	Perceptual/controlled size validation
O3: Two language tasks	Implemented; pilot task completion recorded	Human-validated speech quality only if claimed
O4: Gated AR	Implemented; 17/25 full pilot AR placements	Final merged APK regression across devices
O5: Research evaluation	25 pilot sessions; descriptive mean improvement	Paired scores, control condition and retention follow-up

Table 9.1: Concluding objective assessment with unambiguous evidence labels

9.3 Individual research and product contributions

The research contribution is an assessable learner-originated vocabulary pathway, not a new universal AI model. The central mechanisms are a semantic anchor established by sketch recognition, separation of classifier-normalized input from original sketch-size mapping, independent spelling and target-speech task completion, task-count-driven model transparency and conditional real-world visualization. The product contribution consists of Unity controllers, prefab organisation, child-friendly screen transitions, offline speech and inference dependencies, AR interaction and pseudonymous log exports. As part of group R26-IM-005, C4 provides the literacy-sketch pathway while C1–C3 remain the contributions of the other researchers.

9.4 Future directions

Next steps are merged APK regression, verification of Sentis/Vosk resources, microphone and AR operation, and research log export. New studies should record intended object labels, paired learning assessments, speech-quality judgements when relevant, and a suitable control group. Vocabulary expansion and genuine sketch reconstruction remain future work.

REFERENCES

- [1] R26-IM-005 Research Group, “A Learner-Generated Input Framework for AI and AR-Enhanced Primary Education: Design and Pilot Evaluation,” final group ICAC research-paper manuscript, 2026. [Supplied project research document; verify publication status before submission.]
- [2] S.-H. Zhang, Y.-C. Guo, and Q.-W. Gu, “Sketch2Model: View-aware 3D modeling from single free-hand sketches,” in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2021, pp. 6012–6021. doi: 10.1109/CVPR46437.2021.00595.
- [3] L. Zhang, “Hand-drawn sketch recognition with a double-channel convolutional neural network,” *EURASIP Journal on Advances in Signal Processing*, vol. 2021, art. 73, 2021. doi: 10.1186/s13634-021-00752-4.
- [4] A. K. Bhunia, P. N. Chowdhury, Y. Yang, T. M. Hospedales, T. Xiang, and Y.-Z. Song, “Vectorization and rasterization: Self-supervised learning for sketch and handwriting,” in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2021, pp. 5672–5681.
- [5] E. Korosidou, “The effects of augmented reality on very young learners’ motivation and learning of the alphabet and vocabulary,” *Digital*, vol. 4, no. 1, pp. 195–214, 2024. doi: 10.3390/digital4010010.
- [6] J. Belda-Medina and V. Marrahi-Gomez, “The impact of augmented reality (AR) on vocabulary acquisition and student motivation,” *Electronics*, vol. 12, no. 3, art. 749, 2023. doi: 10.3390/electronics12030749.
- [7] H. Lin and W. Wei, “A systematic review on vocabulary learning in AR and VR gamification context,” *Computers & Education: X Reality*, vol. 4, art. 100057, 2024. doi: 10.1016/j.cexr.2024.100057.
- [8] H. Weigand, P. Johannesson, and B. Andersson, “An artifact ontology for design science research,” *Data & Knowledge Engineering*, vol. 133, art. 101878, 2021. doi: 10.1016/j.datak.2021.101878.
- [9] S. G. Sutton, V. Arnold, P. Collier, and S. A. Leech, “Leveraging the synergies between design science and behavioral science research methods,” *International Journal of Accounting Information Systems*, vol. 43, art. 100536, 2021. doi: 10.1016/j.accinf.2021.100536.
- [10] H.-Y. Chang, T. Binali, J.-C. Liang, G.-L. Chiou, K.-H. Cheng, S. W.-Y. Lee, and C.-C. Tsai, “Ten years of augmented reality in education: A meta-analysis of (quasi-) experimental studies to investigate the impact,” *Computers & Education*, vol. 191, art. 104641, 2022. doi: 10.1016/j.compedu.2022.104641.
- [11] E. L.-C. Law and M. Heintz, “Augmented reality applications for K-12 education: A systematic review from the usability and user experience perspective,” *International Journal of Child-Computer Interaction*, vol. 30, art. 100321, 2021. doi: 10.1016/j.ijcci.2021.100321.
- [12] Alpha Cephei, “Vosk Speech Recognition Toolkit,” Vosk API official repository. Available: <https://github.com/alphacep/vosk-api> (accessed Oct. 2026).
- [13] Unity Technologies, “Sentis overview,” Unity Sentis Documentation. Available: <https://docs.unity.cn/Packages/com.unity.sentis%401.4/manual/index.html> (accessed Oct. 2026).
- [14] Google Creative Lab, “The Quick, Draw! Dataset,” GitHub repository (dataset documentation). Available: <https://github.com/googlecreativelab/quickdraw-dataset> (accessed Oct. 2026).
- [15] Unity Technologies, “AR Foundation,” Unity Package Documentation, version 5.1. Available: <https://docs.unity.cn/Packages/com.unity.xr.arfoundation%405.1/manual/index.html> (accessed Oct. 2026).

- [16] N. N. Wickramasingha, “AR Educational Suite for Primary Students’ Geometry and Literacy: Multisensory Phonics Integration,” individual research project proposal, Sri Lanka Institute of Information Technology, Mar. 2026.
- [17] R26-IM-005 Research Group, “AR Educational Suite for Primary Students’ Geometry and Literacy,” IT4010 Research Project Topic Assessment Form, Sri Lanka Institute of Information Technology, Dec. 2025.

GLOSSARY

Classification confidence: The numeric score assigned by the classifier to a predicted object; not necessarily calibrated probability of correctness.

End-to-end session: The recorded C4 learning interaction from drawing submission to its completion or exit.

Freehand sketch: A child-generated drawing made directly using touch on the digital canvas.

Ground-truth label: The independently established correct or intended class needed to assess classification accuracy.

Progressive reveal: A three-stage transition of a predefined object from low through medium to full visual opacity.

Semantic prefab mapping: Choosing a ready-made model for the recognized object class, instead of reconstructing sketch geometry.

Target-word recognition: Accepting an audio interaction because the speech recognizer returned the expected complete word.

APPENDICES

Appendix A - Research log data dictionary

The following schema reflects the documented C4 logging structure. Its presence in a planned or implemented logger should be verified against the actual final exported files before submission. Field names are included to support reproducible aggregation without exposing child identities. Table A.1 lists the research log data fields.

Field	Meaning
ParticipantID / SessionID	Pseudonymous learner and sequential sketch session identifiers
StartTimestamp / EndTimestamp	Boundaries of the logged attempt
PredictedObject / AIConfidence	Five-class prediction and associated softmax score
SketchSizeRatio	Original drawing-bounds ratio used for relative model scaling
SpellingAttempts / SpellingCompleted	Attempts and acceptance status for exact whole-word spelling
LastSpellingInput SpellingCompletionTimeSeconds	/ Last entry and time to accepted spelling
PronunciationAttempts PronunciationAccepted	/ Speak attempts and target-word recognition acceptance
LastRecognizedSpeech PronunciationCompletionTimeSeconds	/ Recognized hypothesis and completion time
AROpened / ARModelPlaced	AR entry and placement events
TotalSessionTimeSeconds SessionCompleted	/ Overall duration and completed-record flag

Table A.1: C4 anonymised session fields and interpretation

Two variables must NOT be assumed to exist: ExpectedObject and AIPredictionCorrect. The child does not currently specify a separately verified intended object in routine free-draw testing. Therefore the log alone cannot support a child-sketch classification-accuracy numerator. Likewise, PronunciationAccepted is a target-word recognizer outcome and not a validated phonetic quality score.

Appendix B - Suggested learner observation and test instruments

This appendix adapts the proposed observation and feedback themes into a candidate instrument for future ethically approved evaluation. These items are not represented as the exact survey used for the 25-child pilot. A final protocol should be reviewed by educators and the institutional ethics authority, with questions delivered in age-appropriate language. Table B.1 presents a prospective short observation guide.

Item	Possible measure or recording
Can the learner identify where to draw?	Observed without help / with prompt / not completed
Can the learner recover after a sketch misprediction?	Retry count and observed assistance
Can the learner enter or choose the intended word?	Attempts and teacher-observed confusion
Can the learner start and stop the microphone task?	Assistance and environmental disturbance
Can the learner place and manipulate the AR object?	Placement success and touch-control observation
Was the learning activity understandable and enjoyable?	Short age-appropriate response with optional facial scale
What vocabulary did the learner remember later?	Separate consented delayed recall assessment

Table B.1: Candidate short child observation items (not retrospectively claimed)

Appendix C - Example acceptance test procedures

The procedures below are prospective developer checklists to perform against the actual merged Android build. A completed test report must include date, device model, Android version, Unity build identifier, expected and observed output, result, defect identifier and screenshot or console-log evidence. Mark a test as passed only when its evidence is retained.

C.1 Blank and valid drawing. Launch C4, try Continue without drawing, verify the rejection; draw black strokes, save them, then use Clear and verify the texture becomes empty.

C.2 Five-category recognition. Submit representative Apple, Cat, Fish, Sun and Tree drawings. Compare result labels with recorded participant intended-object labels; verify the correct associated model prefab is used.

C.3 Task-state uniqueness. Complete spelling twice while speech remains incomplete and verify the visual model stays in medium opacity. Then complete speech and verify exactly one full-opacity and AR transition.

C.4 Speech lifecycle. With permission allowed, listen and retry repeatedly; with permission denied, confirm a recoverable UI message; navigate Back during listening without leaving an active recording.

C.5 AR and repeat session. After both tasks, place the model on a horizontal plane, rotate, pinch, return to sketch, and create a new drawing; verify a distinct session log and cleared prior flags.

C.6 Asset path regression. Reopen the saved merged scene, inspect no missing scripts, confirm Sentis model asset, prefab references, Vosk StreamingAssets content and single AR infrastructure.

Appendix D - Academic integrity and AI-use disclosure

Draft disclosure for candidate verification (do not sign or submit unchanged): An AI-assisted writing and document-formatting tool was used to organise research materials, draft prose and prepare a structured editable Word file. Source claims were grounded in the individual proposal, the group topic assessment form, the group final research-paper manuscript, the current C4 development context and cited scholarly or official technical sources. The candidate must verify every technical statement, reference, metric, figure and research-ethics assertion against the underlying records, revise unsupported passages and comply with the current SLIIT module policy on generative AI disclosure.