



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE COMPUTACIÓN**

**DESARROLLO DE UN KIT EDUCATIVO BASADO EN VISIÓN POR
COMPUTADOR Y REDES NEURONALES DE APRENDIZAJE PROFUNDO PARA
LA IDENTIFICACIÓN DE CAMPOS LÉXICOS, PROCESOS DE PRELECTURA Y
LECTURA EN EL SOPORTE EDUCATIVO DE NIÑOS DE PRIMERO DE BÁSICA.**

Trabajo de titulación previo a la obtención del
título de Ingeniero en Ciencias de la Computación

AUTOR: SANTIAGO MATEO TORRES CORDERO

TUTOR: ING. VLADIMIR ESPARTACO ROBLES BYKBAEV, PhD.

Cuenca - Ecuador

2024

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Santiago Mateo Torres Cordero con documento de identificación N° 0150095974, manifiesto que:

Soy el autor y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 29 de julio del 2024

Atentamente,

A handwritten signature in blue ink, consisting of several vertical strokes and a series of loops and curves, positioned above a horizontal line.

Santiago Mateo Torres Cordero

0150095974

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Yo, Santiago Mateo Torres Cordero con documento de identificación N° 0150095974, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del Proyecto técnico: “Desarrollo de un kit educativo basado en visión por computador y redes neuronales de aprendizaje profundo para la identificación de campos léxicos, procesos de prelectura y lectura en el soporte educativo de niños de primero de básica.”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 29 de julio del 2024

Atentamente,



Santiago Mateo Torres Cordero

0150095974

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Vladimir Espartaco Robles Bykbaev con documento de identificación N° 0300991817, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DESARROLLO DE UN KIT EDUCATIVO BASADO EN VISIÓN POR COMPUTADOR Y REDES NEURONALES DE APRENDIZAJE PROFUNDO PARA LA IDENTIFICACIÓN DE CAMPOS LÉXICOS, PROCESOS DE PRELECTURA Y LECTURA EN EL SOPORTE EDUCATIVO DE NIÑOS DE PRIMERO DE BÁSICA., realizado por Santiago Mateo Torres Cordero con documento de identificación N° 0150095974, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 29 de julio del 2024

Atentamente,



Ing. Vladimir Espartaco Robles Bykbaev, PhD.

0300991817

AGRADECIMIENTO

Quisiera expresar mi más profundo agradecimiento a todas las personas y entidades que han hecho posible la realización de este trabajo de tesis.

En primer lugar, a mi tutor, el Ing. Vladimir Espartaco Robles Bykbaev, por su invaluable guía, paciencia, y conocimiento. Su apoyo constante y sus consejos han sido fundamentales en cada etapa del desarrollo de este proyecto.

A la Universidad Politécnica Salesiana, por brindarme la oportunidad de formarme y por proporcionarme los recursos necesarios para llevar a cabo esta investigación.

A mis profesores y compañeros de la carrera de Computación, por compartir sus conocimientos y experiencias, y por el constante ánimo y motivación.

A los directivos y docentes del centro educativo “Campus del Saber”, quienes permitieron la implementación y evaluación del kit educativo, y especialmente a los niños que participaron en las pruebas, cuya curiosidad y entusiasmo fueron una fuente constante de inspiración.

A mi familia, por su amor incondicional, su apoyo emocional y financiero, y por siempre creer en mí y en mis capacidades.

Finalmente, a todas aquellas personas que de alguna manera contribuyeron a la realización de este trabajo, mi más sincero agradecimiento.

RESUMEN

Este proyecto describe el desarrollo y la puesta en marcha de un kit educativo basado en visión por computadora y las redes neuronales de aprendizaje profundo, diseñado para mejorar las habilidades de lectura y prelectura en niños de primero de básica en Cuenca, Ecuador. El kit utiliza un modelo YOLOv8 entrenado con un dataset personalizado de pictogramas, junto con técnicas de data augmentation, para detectar y clasificar imágenes en tiempo real. El sistema incorpora una interfaz física con una tabla de dos recuadros y una cámara, proporcionando retroalimentación auditiva inmediata a través de un motor de texto a voz.

La investigación abarcó el estudio de campos léxicos y procesos de prelectura y lectura en el contexto educativo ecuatoriano, el desarrollo del hardware y software del kit, y la evaluación de su efectividad en un entorno escolar real. Se realizaron pruebas con estudiantes y se aplicaron encuestas para medir la percepción y la efectividad del kit.

Los resultados mostraron un alto nivel de interés y comprensión por parte de los niños, con la mayoría expresando un fuerte deseo de utilizar el kit en casa. Sin embargo, se identificaron variaciones en la facilidad de uso percibida, sugiriendo áreas de mejora. El estudio concluye que el kit educativo desarrollado es una herramienta efectiva para apoyar el aprendizaje temprano de la lectura, combinando exitosamente tecnologías avanzadas con métodos pedagógicos.

Palabras clave: Tecnología educativa, Visión por computadora, Prelectura y lectura temprana, YOLOv8, Pictogramas, Data augmentation, Aprendizaje profundo, Educación básica, Inteligencia artificial en educación, Redes neuronales.

ÍNDICE DE CONTENIDO

1.	INTRODUCCIÓN	10
2.	PROBLEMA.....	10
2.1.	Antecedentes	10
2.2.	Importancia y Alcances	10
2.3.	Delimitación	11
3.	OBJETIVOS GENERALES Y ESPECÍFICOS.	12
3.1.	Objetivo General	12
3.2.	Objetivos específicos	12
4.	REVISIÓN DE LA LITERATURA	12
4.1.	Campos Léxicos y Procesos de Prelectura y Lectura.....	12
4.2.	Visión por Computadora y Redes Neuronales en la Educación.....	12
4.3.	Importancia de los Recursos Educativos Interactivos.....	13
4.4.	Tecnologías Utilizadas en el Desarrollo del Kit Educativo.....	13
4.5.	Evaluación del Impacto Educativo de Tecnologías Interactivas	13
5.	MARCO METODOLÓGICO.....	14
5.1.	Investigación Preliminar y Recopilación de Información.....	14
5.1.1.	Estudio de los campos léxicos y procesos de prelectura y lectura:	14
5.1.2.	Análisis de tecnologías de visión por computador y redes neuronales:.....	14
5.2.	Desarrollo del Kit Educativo	14
5.2.1.	Diseño del hardware del kit:	14
5.2.2.	Desarrollo del software del kit:	15
5.3.	Entrenamiento del Modelo.....	15
5.3.1.	Preparación del dataset:	15
5.3.2.	Entrenamiento del modelo YOLOv8:.....	15
5.4.	Implementación del Kit en el Aula	15
5.4.1.	Configuración del entorno de prueba:.....	15
5.4.2.	Ejecución de actividades educativas:	16
5.5.	Evaluación del Kit	16
5.5.1.	Recopilación de datos:	16
6.	RESULTADOS.....	16
6.1.	Resultados del Objetivo Específico 1	16
6.1.1.	Revisión de la literatura y análisis de necesidades educativas:	16
6.2.	Resultados del Objetivo Específico 2	17

6.2.1.	Desarrollo del kit educativo:	17
6.2.2.	Entrenamiento del modelo YOLOv8:.....	20
6.3.	Resultados del Objetivo Específico 3	33
6.3.1.	Implementación del kit en el aula:.....	33
6.3.2.	Pruebas individuales y encuestas:.....	34
7.	CRONOGRAMA	39
8.	PRESUPUESTO	41
9.	CONCLUSIONES	42
10.	RECOMENDACIONES	42
11.	REFERENCIAS BIBLIOGRÁFICAS	43

ÍNDICE DE IMÁGENES

Ilustración 1	Tabla con recuadros para la colocación de pictogramas.	18
Ilustración 2	Trípode con luz y cámara	18
Ilustración 3	Ejemplo del uso del trípode con cámara 1	19
Ilustración 4	Ejemplo del uso del trípode con cámara 2	19
Ilustración 5	Ejemplo simple de las detecciones	20
Ilustración 6	Ejemplo de imágenes para la categoría día	20
Ilustración 7	Código para poner el fondo de la imagen en blanco.	21
Ilustración 8	Ejemplo de las imágenes de diferentes categorías.	22
Ilustración 9	Ejemplo de una nueva imagen.....	23
Ilustración 10	Ejemplo de uso de LabelImg.....	24
Ilustración 11	Ejemplo de la generación de labels para cada imagen.	25
Ilustración 12	etiquetas por categoría.....	26
Ilustración 13	Entrenamiento en Yolov8.....	27
Ilustración 14	Configurado para entrenamiento con CPU.....	27
Ilustración 15	Tiempo por época con CPU	27
Ilustración 16	Configurado para entrenamiento con GPU	27
Ilustración 17	Tiempo por época con GPU	27
Ilustración 18	Métricas del modelo de Yolov8	28
Ilustración 19	Matriz de confusión del entrenamiento	29
Ilustración 20	Ejemplo de predicciones sobre imágenes de validación	30
Ilustración 21	Predicción de resultados sobre la imagen.....	31
Ilustración 22	Motor de texto a voz.....	31
Ilustración 23	Ejemplo de predicciones en tiempo real.....	32
Ilustración 24	Predicción del modelo en el Centro Educativo	34
Ilustración 25	Pruebas de manera individual con diferentes niños.	35
Ilustración 26	Pruebas de manera individual con diferentes niños.	35
Ilustración 27	Resultados de las encuestas.....	36
Ilustración 28	Promedio de los resultados.....	37
Ilustración 29	Resultados de las encuestas.....	38

Ilustración 30 Promedio de los resultados..... 39

1. INTRODUCCIÓN

En la actualidad, es esencial innovar en la educación infantil para mejorar las habilidades de lectura y apoyo académico de los niños. Este informe presenta el desarrollo de un kit educativo que utiliza visión por computadora y redes neuronales de aprendizaje profundo para identificar campos léxicos y procesos de prelectura en niños de primero de básica en Ecuador.

El kit consiste en una tabla con dos recuadros donde los niños colocan pictogramas, que son detectados por una cámara conectada a una computadora que ejecuta Python, OpenCV y un modelo YOLOv8 entrenado. Utilizando imágenes de la plataforma ARASAAC, se creó un dataset con 16 imágenes por categoría y se aplicaron técnicas de data augmentation para entrenar el modelo.

El sistema proporciona retroalimentación auditiva a través de un motor de texto a voz, guiando a los niños en sus tareas. Con ayuda de educadoras se logró contactar con un centro educativo “Campus del Saber” para evaluar la efectividad del kit mediante pruebas individuales y encuestas, demostrando su capacidad para mejorar la lectura y motivar a los niños en el aprendizaje.

2. PROBLEMA

En Cuenca, Ecuador, como en muchas otras partes del mundo, el progreso de las destrezas con respecto a la lectura en los niños de primero de básica enfrenta desafíos significativos. La falta de recursos educativos interactivos y adaptados a las necesidades específicas de estos niños limita su progreso académico y su motivación hacia el aprendizaje. Además, las metodologías tradicionales no siempre logran captar la atención de los estudiantes ni responder a las diversas formas en que los niños aprenden. Este problema requiere una solución innovadora que utilice tecnologías avanzadas para mejorar el proceso educativo.

2.1. Antecedentes

El problema de la alfabetización durante los primeros años de la educación primaria es un tema crítico en Ecuador. Estudios realizados por el Ministerio de Educación de Ecuador han mostrado que un porcentaje significativo de niños en edad escolar tiene dificultades para adquirir habilidades de lectura adecuadas. En Cuenca, esta situación se ve reflejada en los bajos índices de rendimiento académico en las primeras etapas educativas, lo que afecta negativamente su desarrollo académico y social a largo plazo.

La introducción de tecnologías educativas, como la visión por computador y las redes neuronales, ofrece una oportunidad para transformar estos procesos de aprendizaje. Sin embargo, la implementación de estas tecnologías en el contexto educativo ecuatoriano aún es limitada, lo que resalta la necesidad de desarrollar soluciones adaptadas a las condiciones locales y accesibles para los estudiantes y educadores.

2.2. Importancia y Alcances

Abordar este problema es crucial debido al impacto directo dentro del desarrollo académico y personal de los niños de primer grado en Cuenca. Mejorar las habilidades de lectura en esta etapa temprana de la educación no solo favorece a los estudiantes, sino también a las instituciones educativas, al establecer una base estable para la formación y el éxito académico futuro.

- **Sectores afectados:** Este problema afecta principalmente al sector educativo, incluyendo a los estudiantes, maestros y padres de familia. También tiene implicaciones para las instituciones y el sistema educativos en su conjunto.
- **Entidades involucradas:** Las instituciones escolares, así como las organizaciones del sector educativo, se ven directamente afectadas por esta situación. Datos del Instituto Nacional de Estadística y Censos (INEC) y del Ministerio de Educación de Ecuador señalan la urgente necesidad de mejorar las metodologías de enseñanza para obtener mejores resultados en la alfabetización inicial.
- **Contribución del trabajo de grado:** Este trabajo de tesis pretende desarrollar un kit educativo innovador que, mediante el uso de visión por computador y redes neuronales, facilite el aprendizaje de la lectura y la identificación de campos léxicos. Al hacerlo, se espera mejorar el rendimiento académico y la motivación de los estudiantes. Los beneficiarios directos de esta propuesta son los niños de primero de básica en Cuenca, mientras que los beneficiarios indirectos incluyen a sus familias, maestros y la comunidad educativa en general.
- **Documentos oficiales:** Diversos informes y estudios, como los reportes del Ministerio de Educación y las estadísticas del INEC, destacan la importancia de mejorar las habilidades de lectura en los primeros años de educación básica para asegurar un desarrollo académico exitoso.
- **Manifestación del problema:** Actualmente, el problema se manifiesta en los bajos índices de rendimiento académico en las primeras etapas educativas y en la falta de herramientas interactivas y tecnológicas que puedan facilitar el proceso de aprendizaje de la lectura en los niños. Esta carencia limita las oportunidades de los estudiantes para desarrollar habilidades fundamentales de manera efectiva y motivadora.

2.3. Delimitación

- **Geográfica:** Este trabajo se llevará a cabo en la ciudad de Cuenca, Ecuador.
- **Temporal:** La implementación y evaluación del kit educativo se realizará durante el año académico 2024.
- **Sectorial:** El enfoque está en el sector educativo, concretamente en el nivel de primero de básica.

3. OBJETIVOS GENERALES Y ESPECÍFICOS.

3.1. Objetivo General

Desarrollar e implementar un kit educativo basado en visión por computador y redes neuronales de aprendizaje profundo para la identificación de campos léxicos, procesos de prelectura y lectura en el soporte educativo de niños de primero de básica en Ecuador, con el fin de mejorar sus habilidades de lectura y apoyar su desarrollo académico.

3.2. Objetivos específicos

- OE1. Estudiar los principales aspectos de los campos léxicos, procesos de prelectura y lectura dentro del proceso educativo de niños de primero de básica en el contexto ecuatoriano.
- OE2. Desarrollar un kit educativo basado en técnicas de visión artificial y entrenar una red de aprendizaje profundo para la clasificación de pictogramas.
- OE3. Ejecutar un plan de evaluación con ayuda de los educadores para medir el nivel de percepción que tienen los niños con respecto al kit educativo.

4. REVISIÓN DE LA LITERATURA

4.1. Campos Léxicos y Procesos de Prelectura y Lectura

El desarrollo de habilidades de lectura es un proceso gradual que se inicia mucho antes de que los niños comiencen a leer palabras y oraciones completas. Este proceso incluye etapas de prelectura y lectura temprana, donde los niños desarrollan una comprensión básica del lenguaje y sus componentes. Smith y Elley (1994) destacan la importancia de los campos léxicos, que son grupos de palabras relacionadas semánticamente y que ayudan a los niños a entender y organizar el lenguaje de manera más efectiva.

La prelectura implica actividades que desarrollan la conciencia fonológica, el conocimiento del alfabeto y la familiaridad con los libros y las historias (Whitehurst y Lonigan, 1998). Estos fundamentos son cruciales para la transición a la lectura fluida y comprensiva. En el contexto ecuatoriano, se ha destacado la necesidad de enfoques educativos que aborden estos aspectos desde una edad temprana para mejorar las tasas de alfabetización (Ministerio de Educación del Ecuador, 2016).

4.2. Visión por Computadora y Redes Neuronales en la Educación

La integración de tecnologías avanzadas, como la visión por computadora y las redes neuronales, en el ámbito educativo ha mostrado un potencial significativo para mejorar los métodos de enseñanza y aprendizaje. La visión por computadora permite la detección y reconocimiento de objetos en imágenes y videos, lo cual puede ser aplicado para el desarrollo de herramientas educativas interactivas (Szeliski, 2010).

Las redes neuronales convolucionales (CNN) resultan especialmente eficientes en tareas de visión por computadora gracias a su habilidad para aprender características jerárquicas a partir de datos visuales (LeCun, Bengio, & Hinton, 2015). En particular, los modelos YOLO (You Only Look Once) han demostrado ser altamente eficaces con el objetivo de realizar detecciones en tiempo real de diferentes objetos en imágenes, lo que los hace ideales para aplicaciones interactivas en el aula (Redmon et al., 2016).

4.3. Importancia de los Recursos Educativos Interactivos

El uso de recursos educativos interactivos se ha asociado con una mayor motivación y compromiso de los estudiantes, especialmente en la educación primaria (Moreno & Mayer, 2007). La interactividad permite que los estudiantes participen de manera activa en el proceso de aprendizaje, mejorando así la retención de información y el desarrollo de habilidades críticas.

En el contexto de Cuenca, Ecuador, y otras regiones similares, la implementación de tecnologías educativas puede ayudar a superar las limitaciones de recursos tradicionales y proporcionar experiencias de aprendizaje más enriquecedoras y personalizadas (Rodríguez & González, 2017). Esto es particularmente relevante para los niños de primero de básica, quienes se benefician enormemente de las actividades que combinan elementos visuales y táctiles con retroalimentación inmediata.

4.4. Tecnologías Utilizadas en el Desarrollo del Kit Educativo

Python y OpenCV: Python es muy popular en el desarrollo de aplicaciones de visión por computadora por su simplicidad y la gran cantidad de bibliotecas disponibles. OpenCV (Open Source Computer Vision Library) es una librería de código abierto libre que ofrece una amplia variedad de funciones para la manipulación de imágenes y videos, lo que facilita la puesta en funcionamiento de algoritmos de visión por computadora en tiempo real (Bradski & Kaehler, 2008).

YOLOv8: YOLO (You Only Look Once) es una familia de modelos de detección de objetos conocidos por su velocidad y precisión. YOLOv8, una versión avanzada de esta familia permite las detecciones de diferentes objetos en tiempo real con alta precisión, que lo hace ideal para aplicaciones interactivas en entornos educativos (Bochkovskiy, Wang, & Liao, 2020).

4.5. Evaluación del Impacto Educativo de Tecnologías Interactivas

Varias investigaciones han evidenciado que la utilización de tecnologías interactivas en la educación primaria puede influir de manera notable en el desarrollo de habilidades cognitivas y de aprendizaje (Fletcher & Tobias, 2005). En particular, las herramientas basadas en visión por computador y redes neuronales pueden proporcionar retroalimentación inmediata y personalizada, lo que es crucial para el aprendizaje efectivo en los primeros años de educación (Jou & Matarić, 2019).

En el caso de Ecuador, la implementación de estas tecnologías puede ayudar a abordar las desigualdades educativas y proporcionar a los niños de zonas rurales y urbanas acceso a recursos educativos avanzados (Salazar, 2018). La combinación de métodos tradicionales con tecnologías interactivas puede potenciar el aprendizaje y apoyar el desarrollo global de los estudiantes.

5. MARCO METODOLÓGICO.

El marco metodológico de este trabajo de tesis describe detalladamente los pasos seguidos para desarrollar, implementar y evaluar un kit educativo innovador basado en visión por computador y redes neuronales de aprendizaje profundo. Este kit está diseñado para fortalecer las habilidades de lectura y apoyar el desarrollo académico de los niños de primero de básica en Cuenca, Ecuador. A continuación, se detallan los componentes y procesos metodológicos involucrados en el desarrollo del proyecto.

5.1. Investigación Preliminar y Recopilación de Información

5.1.1. Estudio de los campos léxicos y procesos de prelectura y lectura:

- Se llevó a cabo una revisión exhaustiva de la literatura académica sobre los campos léxicos, los procesos de prelectura y lectura, y las metodologías educativas aplicadas en el contexto ecuatoriano.
- Se consultaron fuentes académicas, informes del Ministerio de Educación de Ecuador, y estudios previos para comprender mejor las necesidades educativas y las dificultades enfrentadas por los niños de primero de básica.

5.1.2. Análisis de tecnologías de visión por computador y redes neuronales:

- Se investigaron diversas técnicas de visión por computadora y redes neuronales aplicables al reconocimiento de imágenes y objetos.
- Se evaluaron diferentes modelos y arquitecturas de redes neuronales, seleccionando YOLOv8 por su precisión y eficiencia en la detección de objetos.

5.2. Desarrollo del Kit Educativo

5.2.1. Diseño del hardware del kit:

- Se diseñó una tabla con dos recuadros donde los niños deben colocar los pictogramas. Esta tabla está hecha de un material resistente y fácil de limpiar, adecuado para el uso infantil.

- Se montó una cámara sobre un trípode con luz, apuntando a la tabla para capturar imágenes de los pictogramas colocados por los niños. La cámara se conecta a una computadora mediante un cable USB.

5.2.2. Desarrollo del software del kit:

- Se creó un programa en Python que utiliza OpenCV para capturar y procesar los frames de la cámara en tiempo real.
- Se desarrolló un modelo de reconocimiento de imágenes utilizando YOLOv8. Para ello, se recopiló un dataset de pictogramas desde la página de ARASAAC, que cuenta con pictogramas aprobados por psicólogos. Se seleccionaron varias categorías relevantes y se obtuvieron 16 imágenes por categoría.
- Se aplicó métodos de aumento de datos para incrementar la cantidad y para poder tener así una mayor diversidad de datos de entrenamiento, mejorando la capacidad del modelo para reconocer los pictogramas.

5.3. Entrenamiento del Modelo

5.3.1. Preparación del dataset:

- Se etiquetaron manualmente las imágenes del dataset utilizando herramientas de anotación de imágenes.
- Se dividió el dataset en conjuntos de entrenamiento y validación, asegurando una distribución equilibrada de las categorías.

5.3.2. Entrenamiento del modelo YOLOv8:

- Se configuraron los hiperparámetros del modelo, incluyendo la tasa de aprendizaje, el tamaño del batch y la cantidad de épocas.
- Se entrenó el modelo utilizando un entorno de desarrollo adecuado, aprovechando recursos de procesamiento de GPU para acelerar el entrenamiento.
- Se monitorizaron las métricas de rendimiento durante el entrenamiento, ajustando los hiperparámetros según fuera necesario para optimizar la precisión del modelo.

5.4. Implementación del Kit en el Aula

5.4.1. Configuración del entorno de prueba:

- Se instaló el kit en un centro educativo de Cuenca, en un aula equipada con los recursos necesarios para la implementación (computadora, cámara, trípode, tabla con recuadros). Además se contó con la ayuda de estudiantes de La Universidad Nacional de Educación (UNAE) para validar las diferentes actividades, tiempos para las mismas y para realizar las encuestas.

5.4.2. Ejecución de actividades educativas:

- Se diseñaron varias actividades educativas donde los niños debían colocar los pictogramas en los recuadros según las indicaciones recibidas mediante un sistema de texto a voz.
- El programa de Python capturó los frames de la cámara, procesó las imágenes y reconoció los pictogramas utilizando el modelo YOLOv8. El sistema proporcionó retroalimentación auditiva inmediata a los niños, indicando si sus acciones eran correctas o incorrectas.

5.5. Evaluación del Kit

5.5.1. Recopilación de datos:

- Se realizaron pruebas individuales con los niños, registrando sus interacciones con el kit y sus respuestas a las actividades.
- Se aplicaron encuestas a los niños y a los educadores para medir la percepción y efectividad del kit. Las encuestas incluyeron preguntas sobre la facilidad de uso, la claridad de las instrucciones, la motivación e interés de los niños, y la percepción de mejora en las habilidades de lectura.

6. RESULTADOS.

En esta sección se presentan los resultados alcanzados a lo largo del desarrollo, implementación y evaluación del kit educativo basado en visión por computador y redes neuronales de aprendizaje profundo. Los resultados se organizan en función de los objetivos específicos planteados y se explica cómo estos resultados permitieron cumplir con cada uno de los objetivos.

6.1. Resultados del Objetivo Específico 1

6.1.1. Revisión de la literatura y análisis de necesidades educativas:

La elección de las actividades basadas en pictogramas y sonidos para el kit educativo se fundamentó en una exhaustiva investigación y en las necesidades específicas identificadas en el contexto educativo ecuatoriano. Este proceso incluyó:

- Análisis de Campos Léxicos Relevantes: Tras un repaso de la literatura académica y los informes del Ministerio de Educación de Ecuador, se identificaron los campos léxicos más pertinentes para niños de primero de básica. Junto a educadores estos campos léxicos, como día/noche, playa/montaña, y alimentos saludables/dulces, fueron seleccionados por su relevancia en el desarrollo cognitivo y lingüístico de los niños en esta etapa crucial.
- Enfoque en Habilidades de Prelectura y Lectura: Las actividades fueron diseñadas para abordar las dificultades específicas que los niños enfrentan en el proceso de aprendizaje de la lectura. El uso de pictogramas ayuda a desarrollar la asociación visual-verbal, una habilidad fundamental en la prelectura. La inclusión de elementos auditivos en las actividades 4 y 5 refuerza la conciencia fonológica, otro componente crítico en el desarrollo de habilidades de lectura temprana.
- Integración de Tecnología Interactiva: El dialogo con educadoras de Cuenca resaltaron la necesidad de herramientas tecnológicas motivadoras. El kit educativo responde a esta necesidad al proporcionar una experiencia interactiva que combina estímulos visuales y auditivos, manteniendo el interés de los niños y facilitando un aprendizaje más efectivo.
- Desarrollo de Habilidades Cognitivas: Las actividades de categorización (como separar elementos del día y la noche) ayudan a desarrollar habilidades de clasificación y organización mental, que son fundamentales para la comprensión de conceptos más abstractos en etapas posteriores de la lectura.

6.2. Resultados del Objetivo Específico 2

6.2.1. Desarrollo del kit educativo:

- Se diseñó y fabricó una tabla con dos recuadros para la colocación de pictogramas por parte de los niños como se puede evidenciar en la ilustración 1.

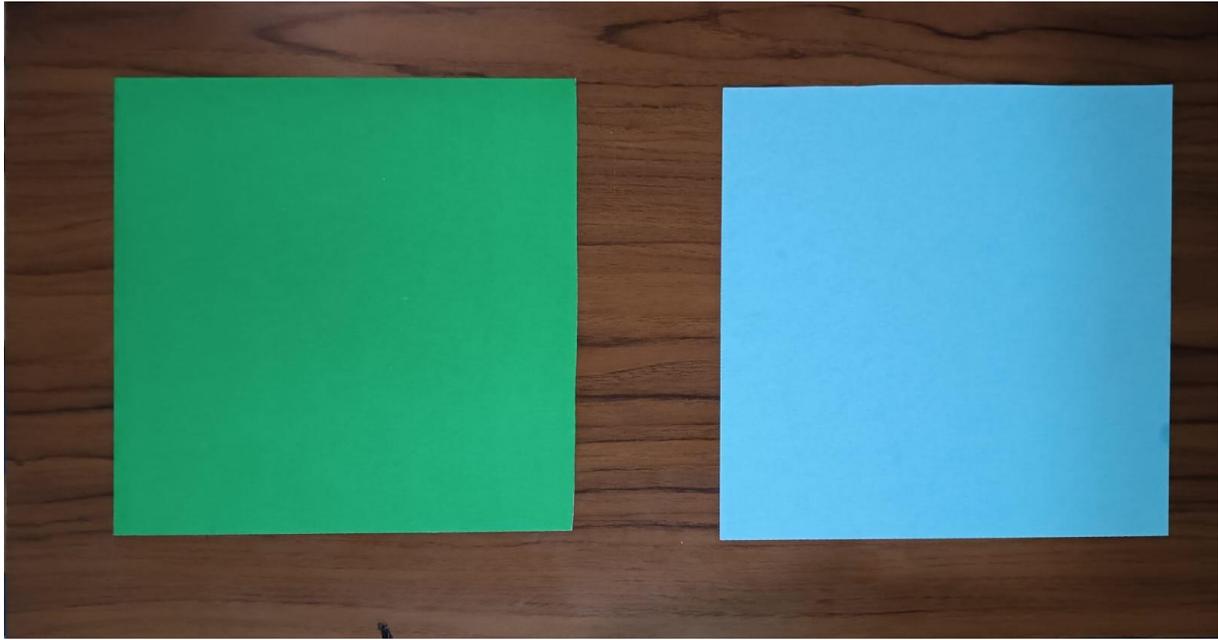


Ilustración 1 Tabla con recuadros para la colocación de pictogramas.

- Con ayuda de un trípode con luz se sostiene una cámara que estará conectada a la computadora.



Ilustración 2 Trípode con luz y cámara



Ilustración 3 Ejemplo del uso del trípode con cámara 1



Ilustración 4 Ejemplo del uso del trípode con cámara 2

- Se desarrolló un programa en Python que utiliza OpenCV para capturar y procesar los frames de la cámara en tiempo real.

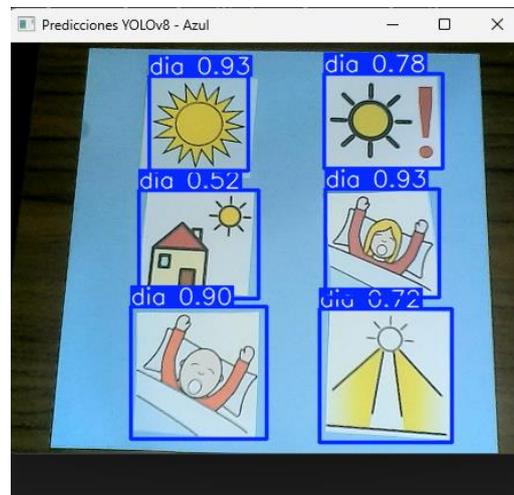


Ilustración 5 Ejemplo simple de las detecciones

6.2.2. Entrenamiento del modelo YOLOv8:

- Se obtuvo un conjunto de imágenes de pictogramas desde la página de ARASAAC, seleccionando varias categorías relevantes y obteniendo 16 imágenes por categoría. En este caso obtuvimos 10 categorías (día, doctor, dulce, granja, mecánico, montaña, noche, playa, saludable, selva).

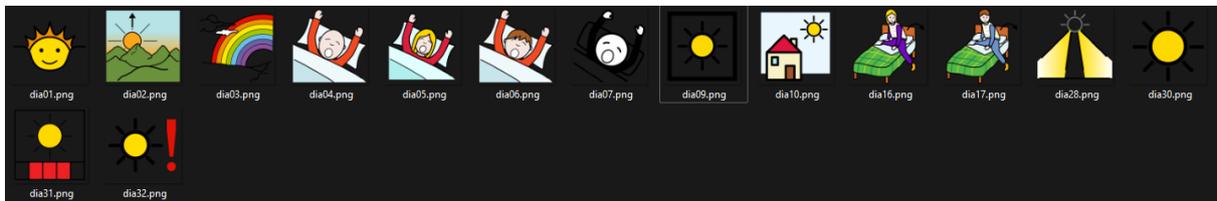


Ilustración 6 Ejemplo de imágenes para la categoría día

- Las imágenes que se descargaban de la página ARASAAC tenían fondo transparente, con ayuda de la librería de OpenCV de Python se puso un fondo blanco ya que había problemas con imágenes que solo tenían objetos negros. El código `make_background_white` utiliza OpenCV para leer una imagen con un canal alfa desde un archivo, separar los canales de color (azul, verde, rojo) y el canal alfa, y luego reemplazar las áreas transparentes (donde el valor del canal alfa es 0) con blanco. Esto se logra mediante el uso de la función `np.where`, que sustituye los valores en los canales de color por blanco en las posiciones donde el canal alfa indica transparencia. Finalmente, los canales modificados se combinan nuevamente, omitiendo el canal alfa, y la imagen resultante se guarda en el archivo especificado. Este proceso garantiza que el fondo transparente de

la imagen se convierta en blanco, al mismo tiempo, el resto de la imagen sigue igual.

```
def fondo_blanco(image_path, output_path):
    # Leer la imagen con transparencia
    img = cv2.imread(image_path, cv2.IMREAD_UNCHANGED)

    # Asegurarse de que la imagen tenga un canal alfa
    if img.shape[2] == 4:
        # Separar los canales
        b, g, r, a = cv2.split(img)
        # Crear una imagen blanca
        white_background = np.ones_like(a, dtype=np.uint8) * 255

        # Usar el canal alfa para mezclar la imagen original con el fondo blanco
        b = np.where(a == 0, white_background, b)
        g = np.where(a == 0, white_background, g)
        r = np.where(a == 0, white_background, r)

        # Combinar los canales de nuevo sin el canal alfa
        img = cv2.merge((b, g, r))

    # Guardar la imagen resultante
    cv2.imwrite(output_path, img)
```

Ilustración 7 Código para poner el fondo de la imagen en blanco.



Ilustración 8 Ejemplo de las imágenes de diferentes categorías.

- Se aplicaron técnicas de data augmentation con el fin de incrementar la cantidad y variedad de datos de entrenamiento. Con ayuda de un script de Python se toma de manera aleatoria las imágenes recolectadas anteriormente y se genera una nueva imagen en la cual se ponen varias imágenes giradas en diferentes ángulos y también cambia un poco de tamaño. En fondo de las imágenes era imágenes de los recuadros de la tabla mostrada anteriormente en la ilustración 1. En este script se podía generar cualquier cantidad de imágenes, para este caso se crearon 400 imágenes, mas las 160 que ya teníamos de antes nos daba un total de 560 imágenes.



Ilustración 9 Ejemplo de una nueva imagen.

- Utilicé la herramienta LabelImg para crear un dataset a partir de mi conjunto de imágenes. Esta herramienta permite etiquetar y anotar imágenes de manera eficiente, generando archivos en formato YOLO que contienen las coordenadas normalizadas de los bounding boxes y las etiquetas correspondientes para cada objeto en las imágenes. Esto hizo más sencilla la preparación de los datos para formar modelos que detecten objetos con YOLO.



Ilustración 10 Ejemplo de uso de LabelImg

Nombre	Fecha de modificación	Tipo	Tamaño
dia01.txt	1/7/2024 16:24	Documento de te...	1 KB
dia02.txt	1/7/2024 16:24	Documento de te...	1 KB
dia03.txt	1/7/2024 16:24	Documento de te...	1 KB
dia04.txt	1/7/2024 16:24	Documento de te...	1 KB
dia05.txt	1/7/2024 16:24	Documento de te...	1 KB
dia06.txt	1/7/2024 16:24	Documento de te...	1 KB
dia07.txt	1/7/2024 16:24	Documento de te...	1 KB
dia09.txt	1/7/2024 16:24	Documento de te...	1 KB
dia10.txt	1/7/2024 16:24	Documento de te...	1 KB
dia16.txt	1/7/2024 16:24	Documento de te...	1 KB
dia17.txt	1/7/2024 16:24	Documento de te...	1 KB
dia28.txt	1/7/2024 16:24	Documento de te...	1 KB
dia29.txt	1/7/2024 16:24	Documento de te...	1 KB
dia30.txt	1/7/2024 16:24	Documento de te...	1 KB
dia31.txt	1/7/2024 16:24	Documento de te...	1 KB
dia32.txt	1/7/2024 16:24	Documento de te...	1 KB
doctor12.txt	1/7/2024 16:24	Documento de te...	1 KB
doctor13.txt	1/7/2024 16:24	Documento de te...	1 KB

Ilustración 11 Ejemplo de la generación de labels para cada imagen.

- Se realizó una separación del 9% para la parte de validación del entrenamiento y el otro 91% para el entrenamiento dándonos un total de 50 imágenes para validación y 510 para entrenamiento. A continuación, podemos ver la cantidad de etiquetas totales para cada categoría en donde podemos ver que la cantidad de todas las etiquetas se encuentra entre las 200 y 250.

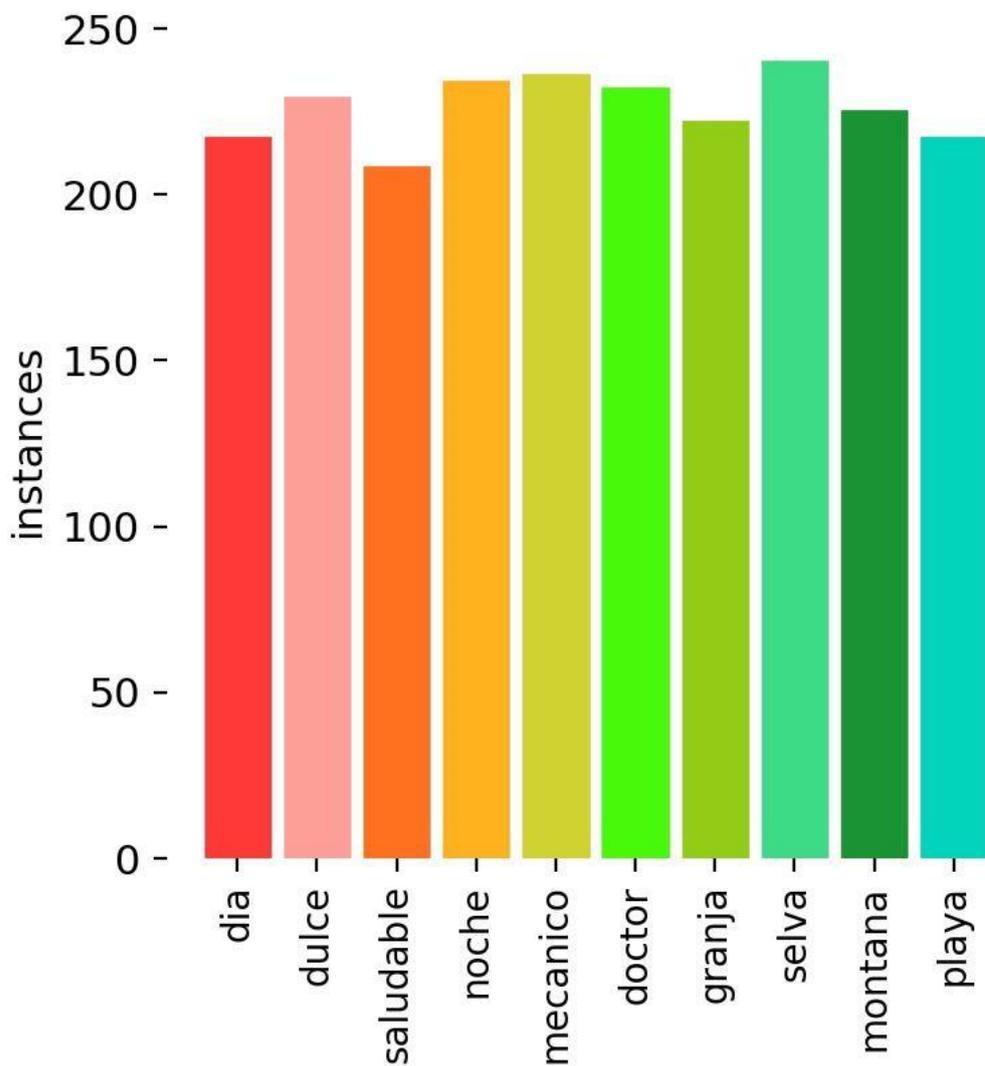


Ilustración 12 etiquetas por categoría

- Se entrenó el modelo YOLOv8 utilizando Python en un entorno de Anaconda, ajustando los hiperparámetros para optimizar la precisión del modelo. Se realizaron varias pruebas con diferentes parámetros del modelo, sin embargo, los parámetros elegidos para los mejores resultados fueron “degress” de 180 grados y “scale” de 0.4. Durante el entrenamiento del modelo YOLOv8, se utilizaron pesos preentrenados derivados de un modelo entrenado en un conjunto de datos general, como COCO. Este enfoque de transferencia de aprendizaje permitió aprovechar el conocimiento previo del modelo sobre características útiles, mejorando la eficiencia del entrenamiento y la precisión del modelo en la tarea específica de detección de pictogramas. La inicialización con pesos

preentrenados facilitó una convergencia más rápida y un rendimiento superior en comparación con el entrenamiento desde cero.

```
>>> results = model.train(data="pictoDataset.yaml", epochs=60, imgsz=640, degrees=180, scale = 0.4)
New https://pypi.org/project/ultralytics/8.2.65 available Update with 'pip install -U ultralytics'
ultralytics YOLOv8.2.28 Python-3.9.19 torch-2.3.1+cu121 CUDA:0 (NVIDIA GeForce RTX 3070, 8192MiB)
engine: trainer: task=detect, mode=train, model=yolov8n.yaml, data=pictoDataset.yaml, epochs=60, time=None, patience=100, batch=16, imgsz=640, save=True, save_period=1, cache=False, device=None, wo
Keras=0, project=None, name=train13, exist_ok=False, pretrained=True, optimizer=auto, verbose=True, seed=0, deterministic=True, single_cls=False, rect=False, cos_lr=False, close_mosaic=10, resume=F
else, amp=True, fraction=1.0, profile=False, freeze=None, multi_scale=False, overlap_mask=True, mask_ratio=4, dropout=0.0, val=True, split_val, save_json=False, save_hybrid=False, conf=None, iou=0.
7, max_det=300, half=False, dnn=False, plots=True, source=None, vid_stride=1, stream_buffer=False, visualize=False, augment=False, agnostic_nms=False, classes=None, retina_masks=False, embed=None,
show=False, save_frames=False, save_txt=False, save_conf=False, save_crop=False, show_labels=True, show_conf=True, line_width=None, format=torchscript, keras=False, optimize=False,
int8=False, dynamic=False, simplify=False, opset=None, workspace=4, nms=False, ltr=0.01, ltrf=0.01, momentum=0.937, weight_decay=0.0005, warmup_epochs=3.0, warmup_momentum=0.0, warmup_bias_lr=0.1,
box7.5, cls=0.5, dfl=1.5, pose=12.0, kobj=1.0, label_smoothing=0.0, nbs=64, hsv_h=0.015, hsv_s=0.7, hsv_v=0.4, degrees=180, translate=0.1, scale=0.4, shear=0.0, perspective=0.0, flipud=0.0, fliplr
=0.5, bgr=0.0, mosaic=1.0, mixup=0.0, copy_paste=0.0, auto_augment=randaugment, erasing=0.4, crop_fraction=1.0, cfg=None, tracker=botsort.yaml, save_dir=runs/detect/train13

from n      params  module                                arguments
0          -1      1      464      ultralytics.nn.modules.conv.Conv          [3, 16, 3, 2]
1          -1      1      4672     ultralytics.nn.modules.conv.Conv          [16, 32, 3, 2]
2          -1      1      7360     ultralytics.nn.modules.block.C2F          [32, 32, 1, True]
3          -1      1      18560    ultralytics.nn.modules.conv.Conv          [32, 64, 3, 2]
4          -1      2      49664    ultralytics.nn.modules.block.C2F          [64, 64, 2, True]
5          -1      1      73984    ultralytics.nn.modules.conv.Conv          [64, 128, 3, 2]
6          -1      2      197632   ultralytics.nn.modules.block.C2F          [128, 128, 2, True]
7          -1      1      295424   ultralytics.nn.modules.conv.Conv          [128, 256, 3, 2]
8          -1      1      469288   ultralytics.nn.modules.block.C2F          [256, 256, 1, True]
9          -1      1      164608   ultralytics.nn.modules.block.SPPF          [256, 256, 5]
10         -1      1      0         torch.nn.modules.upsampling.Upsample      [None, 2, 'nearest']
11         [-1, 0]  1      0         ultralytics.nn.modules.conv.Concat          [1]
12         -1      1      148224   ultralytics.nn.modules.block.C2F          [64, 128, 1]
13         -1      1      0         torch.nn.modules.upsampling.Upsample      [None, 2, 'nearest']
14         [-1, 4]  1      0         ultralytics.nn.modules.conv.Concat          [1]
15         -1      1      37248    ultralytics.nn.modules.block.C2F          [192, 64, 1]
16         -1      1      36992    ultralytics.nn.modules.conv.Conv          [64, 64, 3, 2]
17         [-1, 12] 1      0         ultralytics.nn.modules.conv.Concat          [1]
18         -1      1      123648   ultralytics.nn.modules.block.C2F          [192, 128, 1]
19         -1      1      147712   ultralytics.nn.modules.conv.Conv          [128, 128, 3, 2]
20         [-1, 0]  1      0         ultralytics.nn.modules.conv.Concat          [1]
21         -1      1      493856   ultralytics.nn.modules.block.C2F          [384, 256, 1]
22         [15, 18, 21] 1 753262   ultralytics.nn.modules.head.Detect          [18, [64, 128, 256]]
YOLOv8n summary: 225 layers, 3012798 parameters, 3012782 gradients, 8.2 GFLOPs

Transferred 355/355 items from pretrained weights
TensorBoard: Start with 'tensorboard --logdir runs/detect/train13', view at http://localhost:6006/
Freezing layer 'model.22.dfl.conv.weight'
AMP: running Automatic Mixed Precision (AMP) checks with YOLOv8n...
AMP: checks passed
train: Scanning C:\Users\Estudiante\kiki\yolov8\datasets\pictoDataset\train\labels.cache... 510 images, 0 backgrounds, 0 corrupt: 100% ██████████ 510/510 [00:00<, ?it/s]
val: Scanning C:\Users\Estudiante\kiki\yolov8\datasets\pictoDataset\val\labels.cache... 50 images, 0 backgrounds, 0 corrupt: 100% ██████████ 50/50 [00:00<, ?it/s]
Plotting labels to runs/detect/train13\labels.jpg...
optimizer: 'optimizer=auto' found, ignoring 'lr0=0.01' and 'momentum=0.937' and determining best 'optimizer', 'lr0' and 'momentum' automatically...
optimizer: AdamW(lr=0.000714, momentum=0.9) with parameter groups 57 weight(decay=0.0), 64 weight(decay=0.0005), 63 bias(decay=0.0)
TensorBoard: model graph visualization added
Image sizes 640 train, 640 val
Using 8 dataloader workers
Logging results to runs/detect/train13
```

Ilustración 13 Entrenamiento en YOLOv8

- o Además, se pudo evidenciar la gran diferencia del uso de CPU y GPU en este entrenamiento ya que por el lado de la CPU cada época del entrenamiento llevaba a tardar de 3 a 4 minutos, y en un entrenamiento de 60 épocas no era óptimo, pero con ayuda de la GPU este tiempo se redujo a 3 o 4 segundos

```
>>> results = model.train(data="pictoDataset.yaml", epochs=60, imgsz=640, degrees=180, scale = 0.4, device="cpu")
New https://pypi.org/project/ultralytics/8.2.65 available Update with 'pip install -U ultralytics'
ultralytics YOLOv8.2.28 Python-3.9.19 torch-2.3.1+cu121 CPU (12th Gen Intel Core(TM) i9-12900K)
```

Ilustración 14 Configurado para entrenamiento con CPU

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
1/60	2.32G	1.048	1.234	1.411	146	640: 100% ██████████ 32/32 [04:00<00:00, 7.50s/it]
	Class	Images	Instances	Box(P)	R	mAP50 mAP50-95): 100% ██████████ 2/2 [00:07<00:00, 3.64s/it]

Ilustración 15 Tiempo por época con CPU

```
>>> results = model.train(data="pictoDataset.yaml", epochs=60, imgsz=640, degrees=180, scale = 0.4)
New https://pypi.org/project/ultralytics/8.2.65 available Update with 'pip install -U ultralytics'
ultralytics YOLOv8.2.28 Python-3.9.19 torch-2.3.1+cu121 CUDA:0 (NVIDIA GeForce RTX 3070, 8192MiB)
```

Ilustración 16 Configurado para entrenamiento con GPU

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
1/60	2.35G	1.021	1.238	1.388	131	640: 100% ██████████ 32/32 [00:03<00:00, 9.41it/s]
	Class	Images	Instances	Box(P)	R	mAP50 mAP50-95): 100% ██████████ 2/2 [00:00<00:00, 10.81it/s]

Ilustración 17 Tiempo por época con GPU

- El entrenamiento del modelo YOLOv8 se llevó a cabo durante 60 épocas, mostrando una mejora significativa en el rendimiento a lo largo del proceso. Al inicio del entrenamiento, en la primera época, el modelo presentaba un mAP50 (precisión media al 50% de IoU) de apenas 0.10923, lo que indicaba una baja precisión en la detección de objetos.

A medida que avanzaba el entrenamiento, se observó una rápida mejora en el rendimiento. Para la época 10, el mAP50 ya había alcanzado 0.96647, lo que representa un aumento sustancial en la precisión de detección. Esta tendencia de mejora continuó, aunque de manera más gradual, en las épocas subsiguientes.

Al finalizar el entrenamiento en la época 60, el modelo logró un mAP50 de 0.995, lo que indica una precisión de detección extremadamente alta. Además, se observó una disminución constante en las pérdidas de entrenamiento y validación a lo largo de las épocas, lo que sugiere que el modelo aprendió efectivamente de los datos de entrenamiento sin sobreajustarse. La tasa de aprendizaje también se ajustó gradualmente, comenzando en 0.00022134 y disminuyendo hasta 1.8921e-05 en la última época, lo que probablemente contribuyó a la estabilidad y eficacia del entrenamiento.

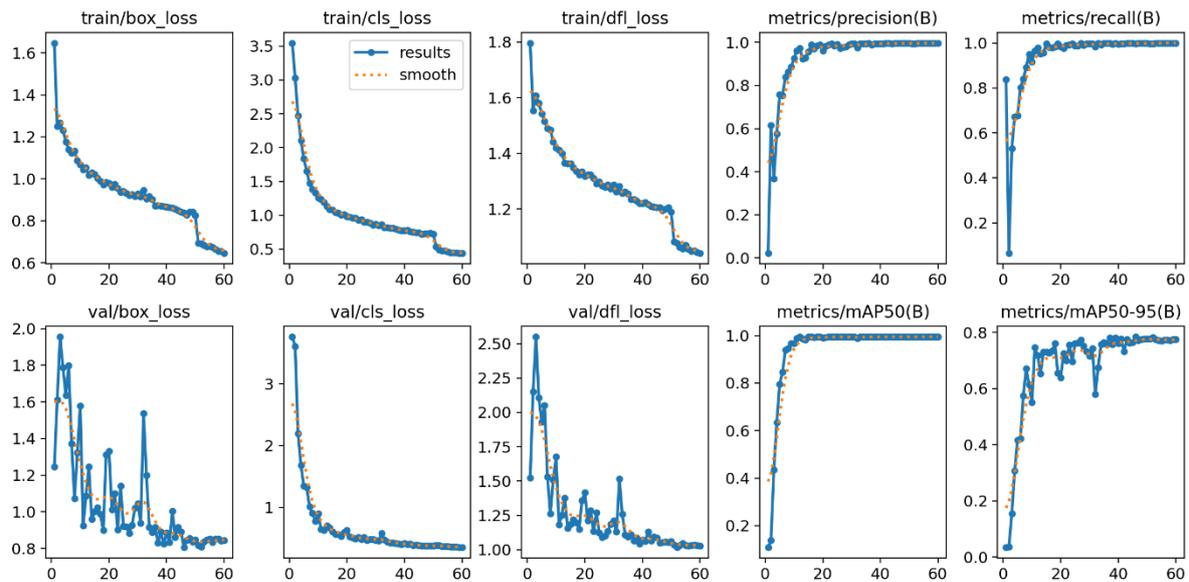


Ilustración 18 Métricas del modelo de Yolov8

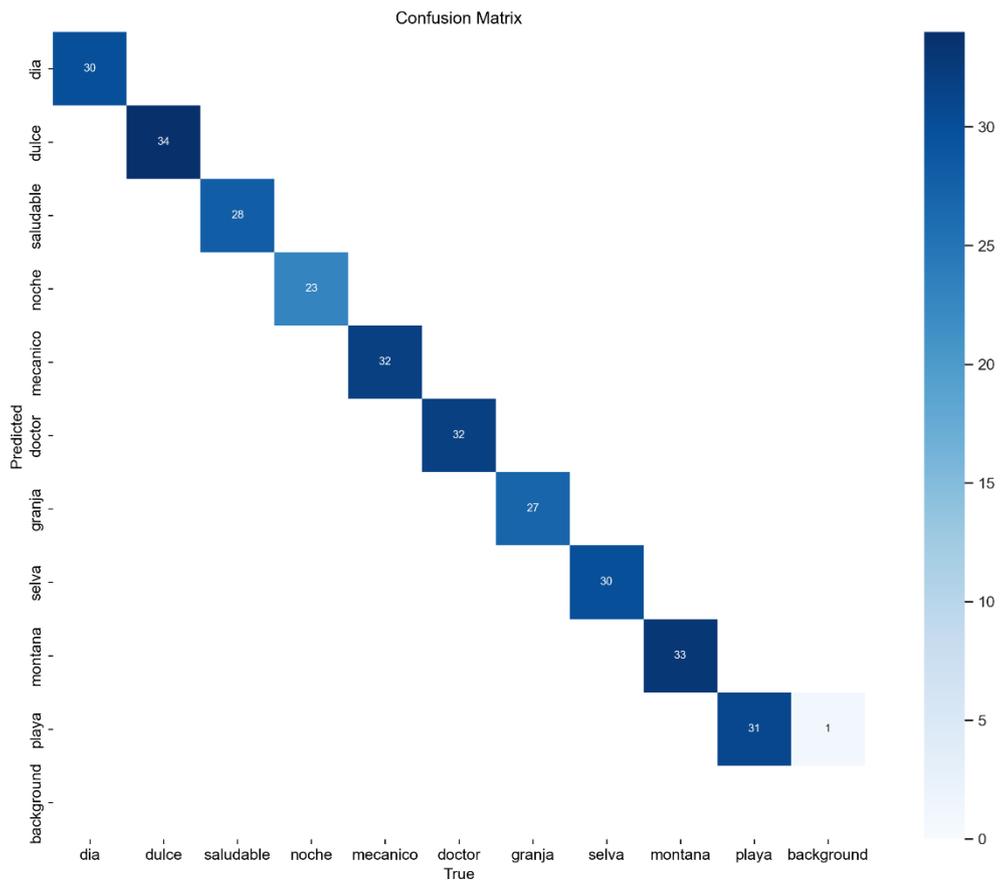


Ilustración 19 Matriz de confusión del entrenamiento

- Podemos visualizar un ejemplo de predicciones hechas sobre las imágenes de predicción en donde se puede evidenciar que la mayoría de las predicciones son correctas.


```

def analizar_imagen(frame_recortado, hablar, del_p, confi):
    resultados = modelo(frame_recortado, conf=confi)
    predicciones = []
    frame_annotado = frame_recortado.copy()

    for resultado in resultados[0].boxes:
        clase = int(resultado.cls[0].item())
        etiqueta = etiquetas[clase]
        if not (del_p and clase == 9): # Excluir "playa" si del_p es True
            predicciones.append(etiqueta)
            box = resultado.xyxy[0].cpu().numpy()
            x1, y1, x2, y2 = map(int, box[:4])
            color = colores[clase]
            cv2.rectangle(frame_annotado, (x1, y1), (x2, y2), color, 2)
            cv2.putText(frame_annotado, etiqueta, (x1, y1 - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.9, color, 2)

    conteo_predicciones = defaultdict(int)
    for prediccion in predicciones:
        conteo_predicciones[prediccion] += 1

    print("Predicciones:", predicciones)
    cv2.imshow('Predicciones YOLOv8 - Azul', frame_annotado)

    if hablar:
        texto_a_voz("Se encontraron las siguientes predicciones:")
        for etiqueta, cantidad in conteo_predicciones.items():
            print(f"Clase: {etiqueta}, Cantidad: {cantidad}")
            texto_a_voz(f"Clase: {etiqueta}, Cantidad: {cantidad}")

```

Ilustración 21 Predicción de resultados sobre la imagen.

```

motor = pyttsx3.init()
def texto_a_voz(texto):
    motor.say(texto)
    motor.runAndWait()
    motor.stop

```

Ilustración 22 Motor de texto a voz.

- Al momento de realizar pruebas con el modelo se pudo evidenciar que el modelo llegaba a tener bastantes falsos positivos con la categoría “playa” y también se pudo evidenciar que la categoría “mecánico” no tenía detecciones.



Ilustración 23 Ejemplo de predicciones en tiempo real.

- Analizando los datos, se observa que algunas categorías fueron clasificadas con buena precisión, mientras que otras presentaron cierta confusión. Por ejemplo, las categorías "playa", "día", "saludable" y "doctor" obtuvieron una clasificación perfecta o casi perfecta, con 6 muestras correctamente identificadas en cada caso. Sin embargo, otras categorías como "Granja", "selva" y "noche" mostraron cierta confusión, con algunas muestras clasificadas erróneamente en otras categorías. Notablemente, la categoría "mecánico" presentó el mayor desafío,

con todas sus muestras clasificadas incorrectamente como "playa". Esta matriz nos dice que el modelo tiene un buen rendimiento en algunas categorías, pero podría beneficiarse de mejoras en la diferenciación entre ciertas clases, especialmente en el caso de "mecánico" y otras categorías con menor precisión.

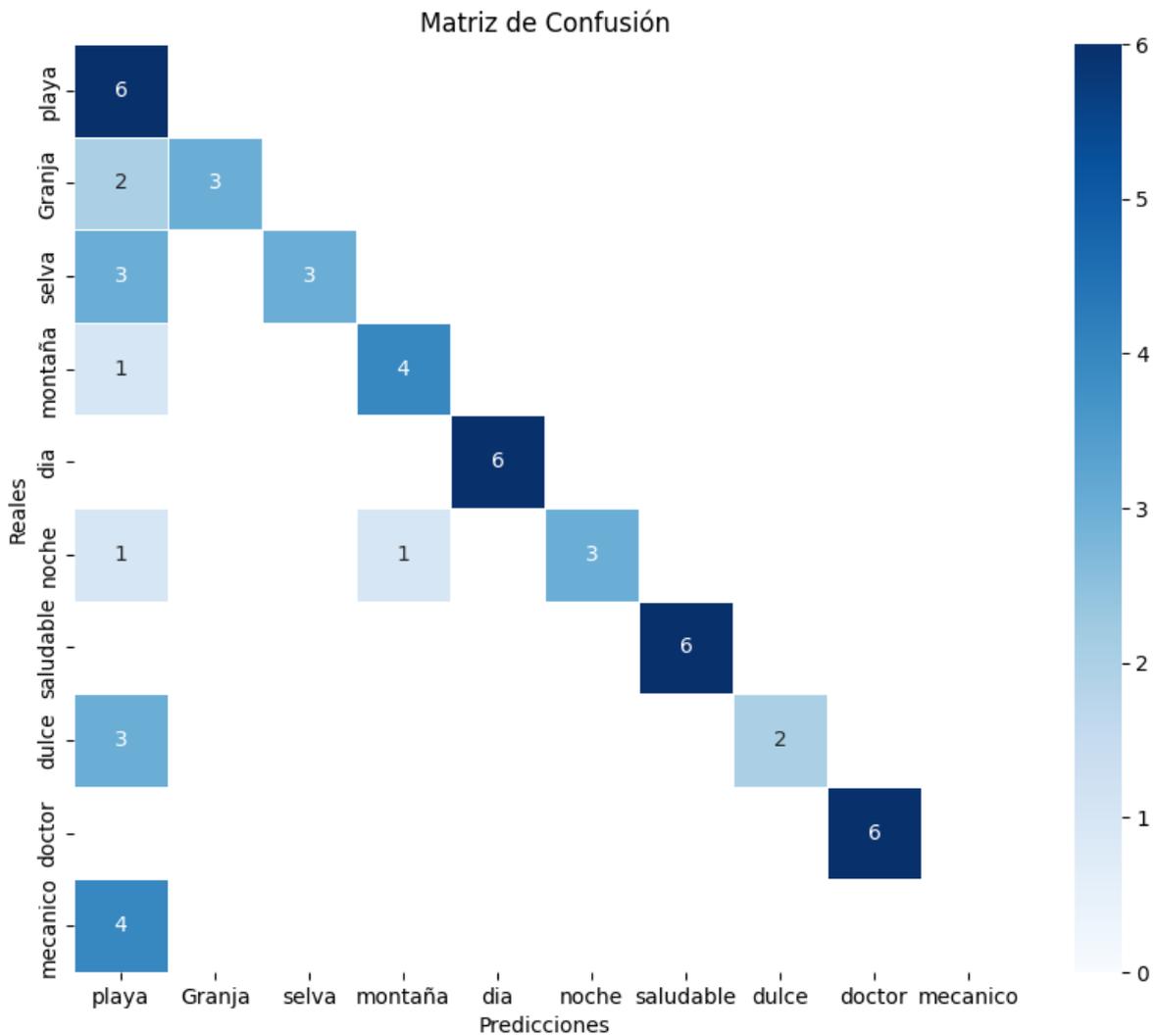


Ilustración 24 Matriz de confusión de las predicciones en tiempo real.

6.3.Resultados del Objetivo Específico 3

6.3.1. Implementación del kit en el aula:

- Se instaló el kit en el centro educativo “Campus del Saber” de Cuenca.



Ilustración 25 Predicción del modelo en el Centro Educativo

- Se llevaron a cabo actividades educativas donde los niños debían colocar los pictogramas en los recuadros según las indicaciones recibidas mediante un sistema de texto a voz. Estas actividades fueron seleccionadas con ayuda de las educadoras y con las estudiantes de la UNAE se determinó un tiempo de dos minutos para cada una de las tres primeras actividades y un minuto para las actividades 4 y 5. Las actividades seleccionadas son:
 1. Colocar los pictogramas referentes a la categoría “día” en un recuadro y los de categoría “noche” en el otro recuadro.
 2. Colocar los pictogramas referentes a la categoría “playa” en un recuadro y los de categoría “montaña” en el otro recuadro.
 3. Colocar los pictogramas referentes a la categoría “dulce” en un recuadro y los de categoría “saludable” en el otro recuadro.
 4. En esta actividad se reproducirá un audio referente a la categoría “selva” y el niño tendrá que solamente escuchando el audio identificar los pictogramas que hacen referencia y ponerlos en el recuadro.
 5. En esta actividad se reproducirá un audio referente a la categoría “granja” y el niño tendrá que solamente escuchando el audio identificar los pictogramas que hacen referencia y ponerlos en el recuadro.

6.3.2. Pruebas individuales y encuestas:

- Se realizaron pruebas individuales con los niños, registrando sus interacciones con el kit y sus respuestas a las actividades.



Ilustración 26 Pruebas de manera individual con diferentes niños.



Ilustración 27 Pruebas de manera individual con diferentes niños.

- Se aplicaron encuestas a los niños y a los educadores para medir la percepción y efectividad del kit. Las encuestas incluyeron preguntas sobre la facilidad de uso, la claridad de las instrucciones y la Se realizaron encuestas a cuatro niños para evaluar su experiencia con el kit educativo. Las preguntas se calificaron en una escala del 1 al 5, donde 5 representa la valoración más positiva. Los resultados obtenidos fueron los siguientes:
1. Interés en el kit (interés): Todos los niños calificaron el kit como totalmente interesante, otorgándole la máxima puntuación de 5.
 2. Comprensión de las actividades: Tres de los cuatro niños entendieron completamente las actividades (puntuación 5), mientras que uno tuvo una comprensión moderada (puntuación 3).
 3. Facilidad de uso (facilidad): Las opiniones variaron, con puntuaciones de 5, 5, 4 y 2, lo que sugiere que la mayoría encontró el kit fácil de usar, pero uno tuvo algunas dificultades.
 4. Comprensión de los pictogramas: Todos los niños entendieron perfectamente los pictogramas, otorgando una puntuación de 5.
 5. Percepción de la velocidad de los juegos (Rápido): Las opiniones variaron significativamente, con puntuaciones de 5, 4, 3 y 2.

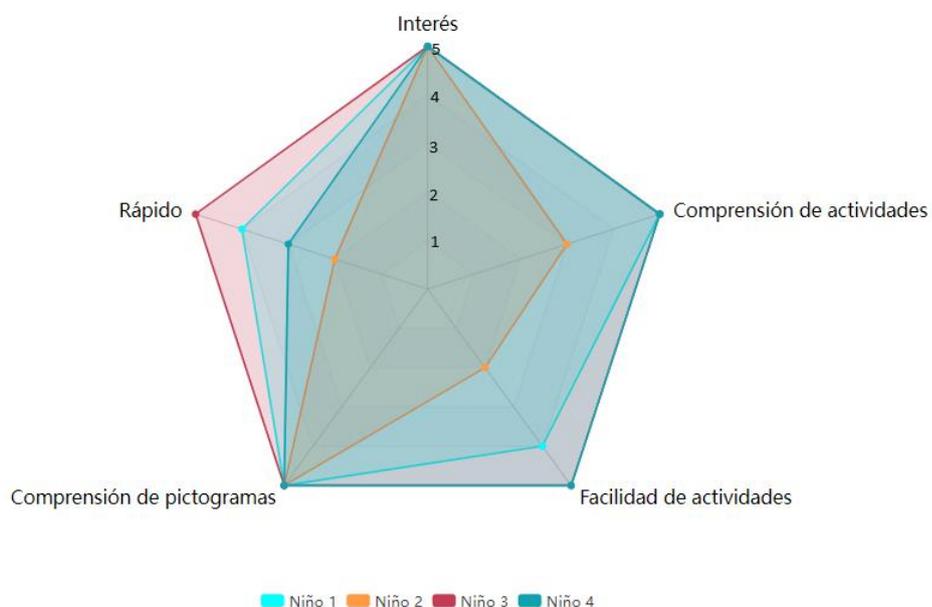


Ilustración 28 Resultados de las encuestas.

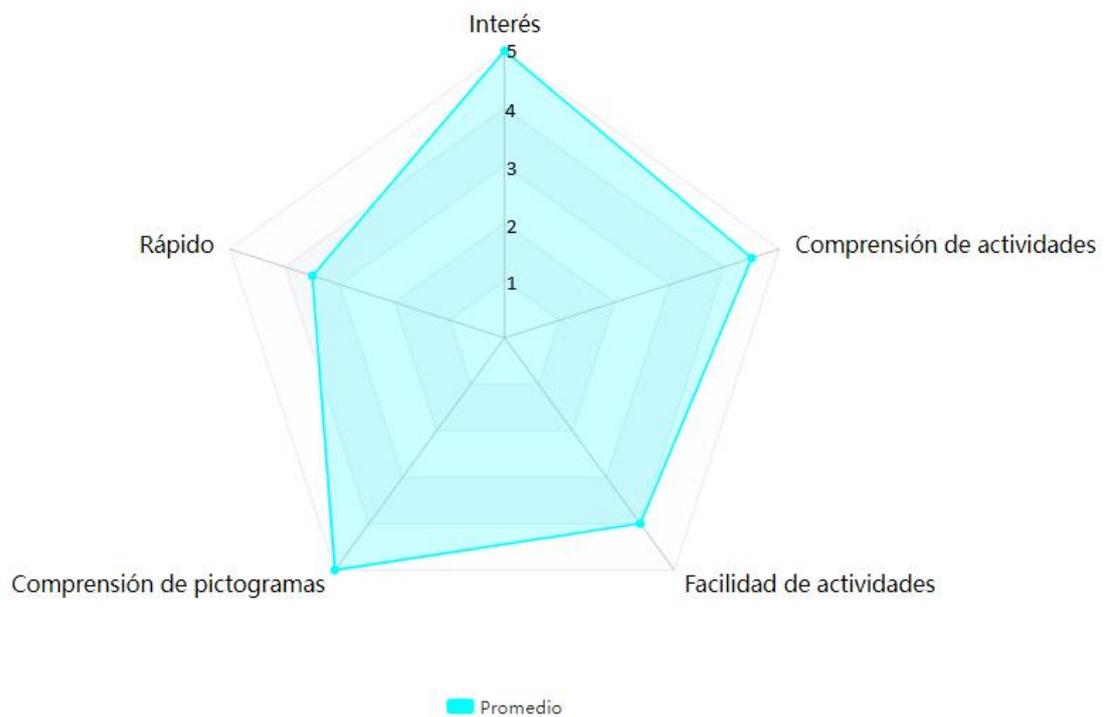


Ilustración 29 Promedio de los resultados

6-10. Facilidad de las actividades individuales:

6. Actividad 1: Puntuaciones variadas (5, 4, 3, 2)
7. Actividad 2: Puntuaciones moderadas a altas (4, 4, 4, 3)
8. Actividad 3: Puntuaciones bajas a moderadas (5, 3, 3, 2)
9. Actividad 4: Puntuaciones bajas a moderadas (4, 3, 2, 1)
10. Actividad 5: Puntuaciones bajas a moderadas (4, 3, 3, 2)

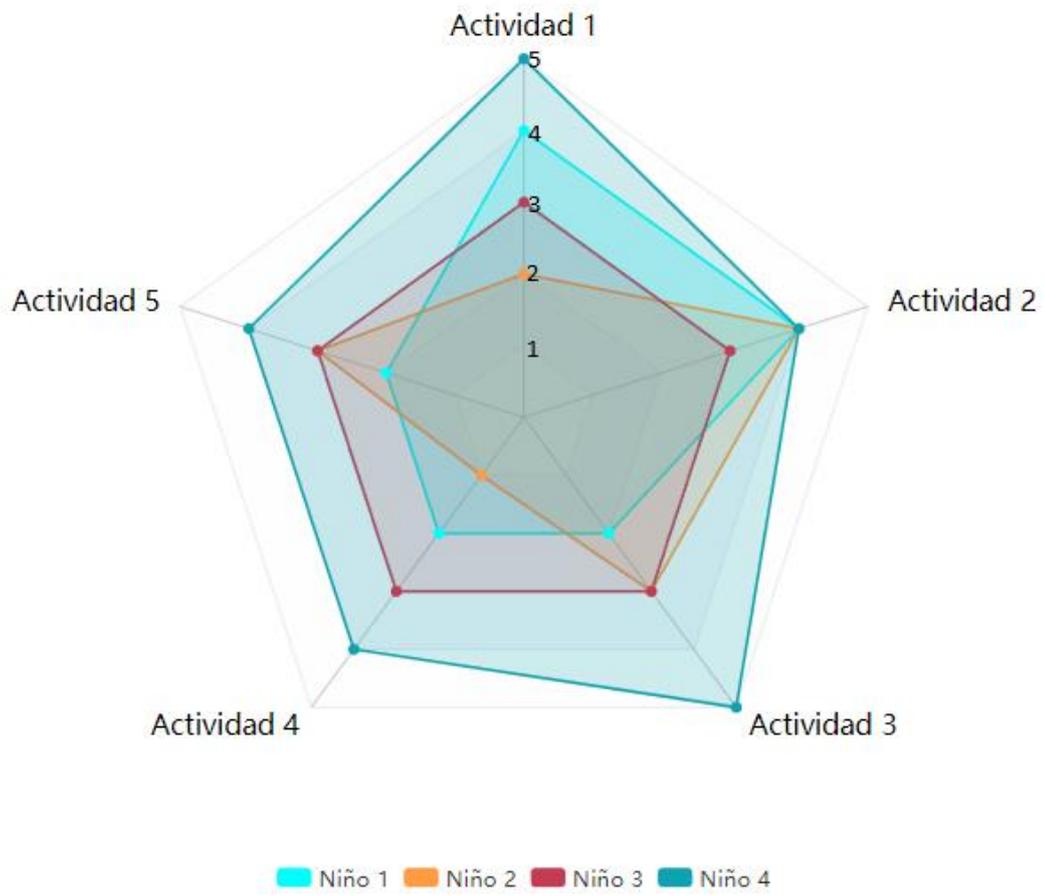


Ilustración 30 Resultados de las encuestas.

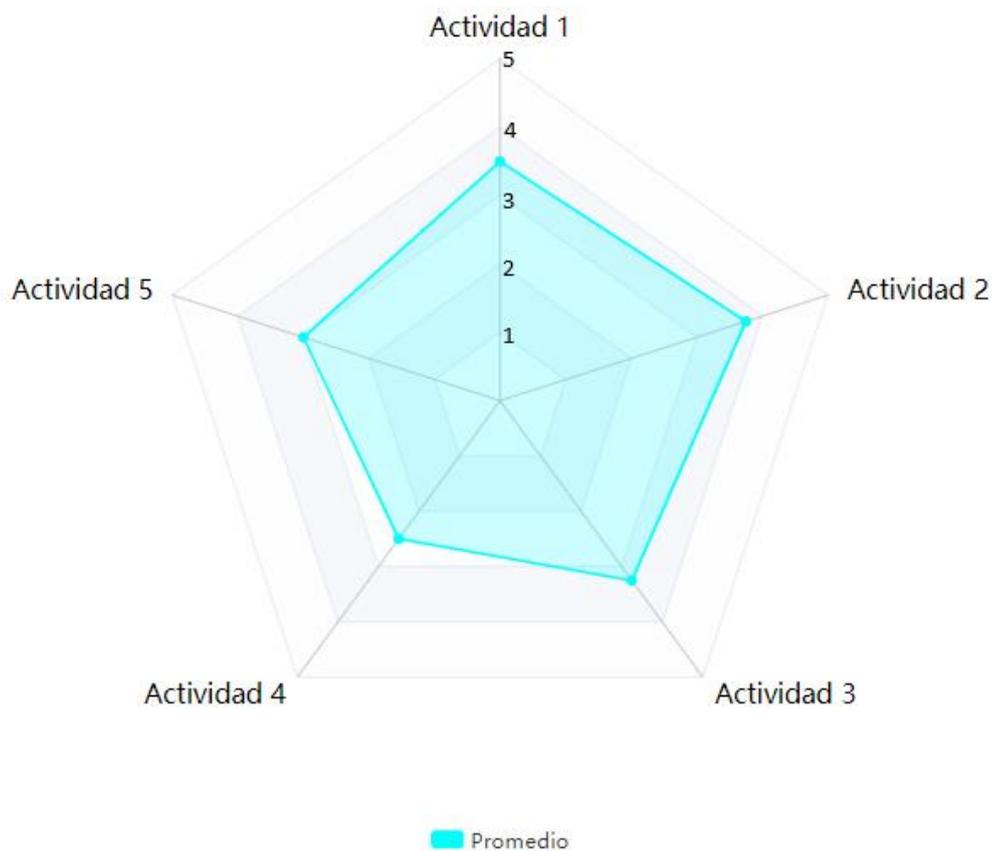


Ilustración 31 Promedio de los resultados

11. Deseo de tener el juego en casa: Todos los niños expresaron un fuerte deseo de tener el juego en casa, calificándolo con 5.
12. Actividad preferida: Tres de los cuatro niños eligieron la actividad relacionada con "Comida" como su favorita, mientras que uno prefirió la actividad "Día y noche".

7. CRONOGRAMA

OE1. Estudiar los principales aspectos de los campos léxicos, procesos de prelectura y lectura dentro del proceso educativo de niños de primero de básica en el contexto ecuatoriano.

1. Estudiar los aspectos principales relacionados a los campos léxicos.
2. Estudiar los aspectos principales relacionados a los procesos de prelectura.
3. Estudiar los aspectos principales relacionados a los procesos de lectura.

OE2. Desarrollar un kit educativo basado en técnicas de visión artificial y entrenar una red de aprendizaje profundo para la clasificación de pictogramas.

1. Realizar la estructura del kit para el soporte de la cámara
2. Diseñar el programa.
3. Recolectar imágenes y entrenar modelos.

OE3. Ejecutar un plan de evaluación con ayuda de los educadores para medir el nivel de percepción que tienen los niños con respecto al kit educativo.

1. Diseñar el cuestionario para la evaluación.
2. Realizar pruebas piloto del programa y recopilar resultados.
3. Analizar los resultados obtenidos

Objetivo	Actividad	Fecha Inicio	Fecha Fin	Horas por responsable	Responsables
OE1	Ac. 1	29/4/2024	29/4/2024	4	Santiago Torres
OE1	Ac. 2	30/4/2024	30/4/2024	4	Santiago Torres
OE1	Ac. 3	1/4/2024	1/5/2024	4	Santiago Torres
OE2	Ac. 1	2/5/2024	15/5/2024	40	Santiago Torres
OE2	Ac. 2	16/5/2024	24/5/2024	24	Santiago Torres
OE2	Ac. 3	27/5/2024	21/7/2024	100	Santiago Torres

OE3	Ac. 1	24/06/2024	25/06/2024	8	Santiago Torres
OE3	Ac. 2	26/6/2024	2/07/2024	8	Santiago Torres
OE3	Ac. 3	3/07/2024	5/07/2024	8	Santiago Torres

8. PRESUPUESTO

DENOMINACIÓN	CANT.	COSTO UNITARIO	COSTO TOTAL
	Unidades	Dólares	Dólares
1. Bienes			
Papel Bond A-4	2	10.00	20.00
Copias	100	0.05	5.00
Estructura del Kit	1	400.00	400.00
2. Tecnológico			
Computadora portátil	1	1000.00	1,000.00
Cámara	1	35.00	35.00
2. Servicios			
Servicios de Internet	6	28.00	168.00
Material Bibliográfico	7	35.00	245.00
4. Personal			

Estudiante/Desarrollador	240 horas	20 por hora	4,800.00
3. Otros			
Imprevistos	1	100.00	100.00
TOTAL			7,078.00

9. CONCLUSIONES

- El kit educativo desarrollado demuestra ser una herramienta para mejorar las habilidades de lectura y prelectura en niños de primero de básica, combinando exitosamente tecnologías avanzadas como la visión por computador y las redes neuronales con métodos pedagógicos.
- La implementación de un modelo YOLOv8 entrenado con un dataset personalizado de pictogramas resultó ser eficaz para la detección y clasificación en tiempo real.
- Las técnicas de aumento de datos aplicadas en el proceso de entrenamiento del modelo contribuyeron significativamente a mejorar su rendimiento y precisión, especialmente considerando el tamaño limitado del dataset inicial.
- La retroalimentación auditiva inmediata proporcionada por el sistema de texto a voz demostró ser un elemento clave para mantener la motivación e interés de los niños durante las actividades educativas.
- Los resultados de las encuestas indican un alto nivel de aceptación y entusiasmo por parte de los niños hacia el kit educativo, con la mayoría de ellos expresando un fuerte deseo de tener acceso al juego en casa.
- Se observó una variación en la facilidad de uso percibida entre los diferentes usuarios, lo que sugiere la necesidad de ajustes adicionales para mejorar la accesibilidad del kit para todos los niños.

10. RECOMENDACIONES

- Realizar un estudio longitudinal para evaluar el impacto a largo plazo del kit educativo en el fortalecimiento de habilidades de lectura y prelectura de los niños.
- Expandir el dataset de pictogramas y considerar la inclusión de nuevas categorías para ampliar el alcance educativo del kit.
- Implementar técnicas de fine-tuning en el modelo YOLOv8 para abordar los problemas identificados con falsos positivos en la categoría "playa" y falsos negativos en la categoría "mecánico".
- Desarrollar versiones adaptadas del kit para diferentes grupos de edad y niveles educativos, permitiendo una continuidad en el uso del kit a lo largo del proceso de aprendizaje.

- Incorporar un sistema de seguimiento del progreso individual de cada niño, permitiendo a los educadores identificar áreas de mejora y personalizar las actividades según las necesidades específicas de cada uno de los estudiantes.
- Explorar la posibilidad de crear una versión del kit que pueda ser utilizada en dispositivos móviles o tabletas, aumentando así su accesibilidad y permitiendo su uso en entornos domésticos.
- Realizar talleres de capacitación para educadores sobre el uso efectivo del kit en el aula, maximizando su potencial como herramienta de apoyo educativo.
- Considerar la inclusión de elementos de gamificación adicionales para aumentar aún más la motivación y el compromiso que tendrían los niños con respecto a las actividades educativas.

11. REFERENCIAS BIBLIOGRÁFICAS

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*. <https://arxiv.org/abs/2004.10934>

Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer Vision with the OpenCV Library*. Sebastopol, CA: O'Reilly Media.

Fletcher, J. D., & Tobias, S. (2005). The multimedia principle. In R. E. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning* (pp. 117-133). Cambridge University Press.

Jou, B., & Matarić, M. J. (2019). Augmenting elementary school education with online self-paced multimedia content. *Educational Technology Research and Development*, 67(6), 1563-1580. <https://doi.org/10.1007/s11423-019-09667-5>

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444. <https://doi.org/10.1038/nature14539>

Ministerio de Educación del Ecuador. (2016). Informe de resultados de aprendizaje en educación básica. Quito, Ecuador: Ministerio de Educación.

Moreno, R., & Mayer, R. E. (2007). Interactive multimodal learning environments. *Educational Psychology Review*, 19(3), 309-326. <https://doi.org/10.1007/s10648-007-9047-2>

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779-788. <https://doi.org/10.1109/CVPR.2016.91>

Rodríguez, J., & González, M. (2017). Tecnología educativa en la educación primaria: Un estudio de caso en Cuenca, Ecuador. *Revista Latinoamericana de Tecnología Educativa*, 16(1), 45-59. <https://doi.org/10.12795/rlted.2017.v16.45>

Salazar, P. (2018). Educational technology in Ecuador: Current state and future perspectives. *Journal of Educational Technology & Society*, 21(3), 195-206. <https://www.jstor.org/stable/26458506>

Smith, F., & Elley, W. B. (1994). How children learn to read: Insights from the New Zealand experience. *The Reading Teacher*, 48(3), 222-225. <https://doi.org/10.1598/RT.48.3.7>

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. London, England: Springer.

Whitehurst, G. J., & Lonigan, C. J. (1998). Child development and emergent literacy. *Child Development*, 69(3), 848-872. <https://doi.org/10.2307/1132208>