

**PROPUESTA DE APOYO A LA EDUCACIÓN INFANTIL REFERENTE A LA  
PERCEPCIÓN DE ALGUNOS ATRIBUTOS BÁSICOS EN FIGURAS  
GEOMÉTRICAS Y EL RECONOCIMIENTO DE PATRONES SIMPLES  
MEDIANTE LA VISIÓN ARTIFICIAL Y EL JUEGO EDUCATIVO**



**CARLOS ENRIQUE REALPE GÓMEZ**

**UNIVERSIDAD DEL CAUCA**

**FACULTAD DE CIENCIAS NATURALES, EXACTAS Y DE LA EDUCACIÓN**

**DEPARTAMENTO DE FÍSICA**

**INGENIERÍA FÍSICA**

**POPAYÁN**

**2019**

**PROPUESTA DE APOYO A LA EDUCACIÓN INFANTIL REFERENTE A LA  
PERCEPCIÓN DE ALGUNOS ATRIBUTOS BÁSICOS EN FIGURAS  
GEOMÉTRICAS Y EL RECONOCIMIENTO DE PATRONES SIMPLES  
MEDIANTE LA VISIÓN ARTIFICIAL Y EL JUEGO EDUCATIVO**

**CARLOS ENRIQUE REALPE GÓMEZ**

**Trabajo en modalidad de investigación presentado como requisito para  
optar al título de Ingeniero Físico**

**Director**

**Ing. EDGAR LEONAIRO PENCUE FIERRO**

**UNIVERSIDAD DEL CAUCA**

**FACULTAD DE CIENCIAS NATURALES, EXACTAS Y DE LA EDUCACIÓN**

**DEPARTAMENTO DE FÍSICA**

**INGENIERÍA FÍSICA**

**POPAYÁN**

**2019**

## NOTA DE ACEPTACIÓN

---

---

---

---

---

---

---

---

Firma del director

---

Firma del jurado

---

Firma del jurado

Popayán, marzo de 2019.

## DEDICATORIA

Este trabajo está dedicado a los niños de la primera infancia que empiezan a percibir el mundo que los rodea y a las personas que de buen corazón buscan los métodos para enseñar y suplir las necesidades pedagógicas que imperan en la vida de estos pequeños que se convertirán en las generaciones del futuro.

*“Instruye al niño en su camino, y aun cuando fuere viejo no se apartará de él”*

Proverbios 22:6

## **AGRADECIMIENTOS**

Se agradece primeramente a Dios por la vida, la salud y el apoyo para realizar este trabajo, a mi familia que brindó sustento y motivación para continuar estudiando, al Hogar de Bienestar Familiar los Jilgueros por permitir realizar las pruebas lúdicas necesarias para este trabajo, a todo el grupo docente que desde la infancia me brindaron la educación que me permitió llegar hasta este punto y a todos mis compañeros que fueron de gran apoyo para superar los retos del mundo académico; Dios los guarde y bendiga a todos.

## CONTENIDO

	pág.
RESUMEN.....	12
1. INTRODUCCIÓN .....	13
2. PLANTEAMIENTO DEL PROBLEMA .....	14
3. JUSTIFICACIÓN .....	15
4. OBJETIVOS .....	16
4.1. OBJETIVO GENERAL.....	16
4.2. OBJETIVOS ESPECÍFICOS.....	16
5. MARCO TEÓRICO.....	17
5.1. EL JUEGO EDUCATIVO .....	17
5.1.1. Algunos teóricos relevantes del juego educativo .....	17
5.1.2. El tangram y sus beneficios .....	19
5.1.3. Las TIC en la educación temprana.....	20
5.1.4. La visión por computador y juego educativo .....	21
5.2. PROCESAMIENTO DIGITAL DE IMÁGENES.....	22
5.2.1. Aplicaciones .....	22
5.2.2. Características básicas de una imagen digital .....	23
5.2.3. Etapas del procesamiento.....	24
5.3. RECONOCIMIENTO DE FIGURAS GEOMÉTRICAS .....	25
5.3.1. OpenCV .....	25
5.3.2. Algoritmo de Ramer–Douglas–Peucker .....	25
5.4. CARACTERÍSTICAS DE UNA CÁMARA DIGITAL .....	28
5.4.1. Elementos de una cámara .....	29
5.4.2. Parámetros a tener en cuenta.....	31
6. PROCEDIMIENTO EXPERIMENTAL .....	32
6.1. ENTORNO FÍSICO.....	32
6.1.1. Configuración de la cámara .....	33
6.1.2. Iluminación.....	34
6.1.3. Características para el fondo .....	35
6.1.4. Características para las figuras.....	35
6.2. DESARROLLO DE LAS ACTIVIDADES LÚDICAS .....	37
6.2.1. Lúdicas para identificar colores.....	37
6.2.2. Lúdicas para identificar tamaños.....	39
6.2.3. Lúdica para identificar figuras .....	40
6.2.4. Componente de direccionalidad.....	42
6.2.5. Lúdica para identificar patrones simples .....	43
6.2.6. Percepción plana espacial y el tangram.....	45
6.3. DESARROLLO DE LA APLICACIÓN SOFTWARE .....	45
6.3.1. Entorno de desarrollo.....	45
6.3.2. Arquitectura de la aplicación .....	46

6.3.2.1.	Muestreo.....	47
6.3.2.2.	Componente gráfico .....	48
6.3.2.3.	Funcionalidad de la aplicación.....	49
6.3.2.4.	Procesamiento digital de imágenes .....	51
6.3.3.	Interfaz gráfica de usuario.....	62
7.	RESULTADOS Y ANÁLISIS.....	64
7.1.	RESULTADOS DEL PROCESAMIENTO DIGITAL DE IMÁGENES ...	64
7.1.1.	Reconocimiento de formas, tamaños y colores.....	64
7.1.2.	Reconocimiento de la flecha y su direccionalidad .....	66
7.2.	RESULTADOS REFERENTES AL COMPONENTE VISUAL DE LA APLICACIÓN.....	68
7.2.1.	Interfaz gráfica de usuario.....	68
7.2.2.	Lúdicas.....	69
7.2.2.1.	Lúdicas con colores .....	70
7.2.2.2.	Lúdicas con tamaños.....	71
7.2.2.3.	Lúdicas con figuras.....	72
7.2.2.4.	Lúdicas con patrones simples.....	73
7.2.2.5.	Desarrollo del tangram .....	74
7.3.	RESULTADOS DE IMPLEMENTACIÓN .....	75
7.3.1.	Jugabilidad .....	75
7.3.2.	Mejoras .....	76
7.3.3.	Manejo y aprendizaje .....	77
7.3.4.	Implementación del Tangram.....	79
8.	CONCLUSIONES.....	81
9.	RECOMENDACIONES .....	82
	BIBLIOGRAFÍA.....	83
	ANEXOS.....	85

## LISTA DE TABLAS

	pág.
<b>Tabla 1.</b> Especificaciones para la cámara utilizada en este trabajo. ....	34
<b>Tabla 2.</b> Primer grupo de figuras. ....	35
<b>Tabla 3.</b> Segundo grupo de figuras. ....	36
<b>Tabla 4.</b> Tercer grupo de figuras. ....	36
<b>Tabla 5.</b> Cuarto grupo de figuras. ....	36
<b>Tabla 6.</b> Características generales de la máquina y del software. ....	48
<b>Tabla 7.</b> Tipos de arreglos y sus matrices asociadas. ....	54
<b>Tabla 8.</b> Valoración de conocimientos antes de usar la aplicación. ....	77
<b>Tabla 9.</b> Valoración de conocimientos después de usar la aplicación. ....	78
<b>Tabla 10.</b> Resultados de implementación del tangram. ....	79



## LISTA DE FIGURAS

pág.

<b>Figura 1.</b> Subsistema de manejo de los Bloques Lógicos (Cadillo León, 2012). .....	22
<b>Figura 2.</b> Imagen digital.....	23
<b>Figura 3.</b> Imagen binaria preparada para detectar los contornos o figuras geométricas. ....	26
<b>Figura 4.</b> Implementación del algoritmo de Douglas-Peucker. (Koning, 2011).26	26
<b>Figura 5.</b> Ejemplo de aplicación del algoritmo de Douglas-Peucker. ....	27
<b>Figura 6.</b> Principio físico de una cámara (Fernández, 2013).....	28
<b>Figura 7.</b> Componentes de una cámara (Fernández, 2013). ....	29
<b>Figura 8.</b> Diafragma de una cámara (Fernández, 2013). ....	30
<b>Figura 9. A.</b> Sensor de imagen de una cámara web. <b>B.</b> conversión de la imagen en digital. (Contreras, 2013) (Welleman, 2008).....	30
<b>Figura 10.</b> Ángulo de visión según distancia focal del objetivo de una cámara. Diagrama sobre la relación entre la distancia focal del objetivo y el ángulo de visión abarcado: 1. Objetivo ojo de pez. 2. Objetivo gran angular. 3. Objetivo normal. 4. Teleobjetivo. 5. Súper teleobjetivo (MrSpyDoS, 2012).....	31
<b>Figura 11.</b> Arquitectura general para el desarrollo de la herramienta. ....	32
<b>Figura 12.</b> Esquema para el escenario físico. ....	33
<b>Figura 13.</b> Cámara usada durante la experimentación. ....	34
<b>Figura 14.</b> Esquema de juego con colores. ....	37
<b>Figura 15.</b> Segundo esquema de juego con colores. ....	38
<b>Figura 16.</b> Tercer esquema de juego con colores. ....	38
<b>Figura 17.</b> Esquema de juego con tamaños relativos. ....	39
<b>Figura 18.</b> Segundo esquema de juego con tamaños. ....	39
<b>Figura 19.</b> Tercer esquema de juego con tamaños. ....	40
<b>Figura 20.</b> Esquema de juego con figuras. ....	40
<b>Figura 21.</b> Segundo esquema de juego con figuras. ....	41
<b>Figura 22.</b> Tercer esquema de juego con figuras. ....	41
<b>Figura 23.</b> Esquema para el componente direccional del juego. ....	42
<b>Figura 24.</b> Acciones asociadas a las direcciones de la flecha. ....	42
<b>Figura 25.</b> Fotografía extraída de (Zurira A., 1992). ....	43
<b>Figura 26.</b> Esquema de juego para el reconocimiento de patrones con el color como variable. ....	44
<b>Figura 27.</b> Ejemplo de patrones para la aplicación. ....	44
<b>Figura 28.</b> Esquema de juego con el tangram. ....	45
<b>Figura 29.</b> Esquema general de la aplicación. ....	47
<b>Figura 30.</b> Esquema general de la jerarquía para el desarrollo e implementación de gráficos mediante el lenguaje QML. ....	48

<b>Figura 31.</b> Esquema general para la funcionalidad del juego referente al mando de la aplicación. ....	50
<b>Figura 32.</b> Proceso general del juego o algoritmo principal de la aplicación. ....	51
<b>Figura 33.</b> Flujo del procesamiento de imágenes para cada fotograma. ....	52
<b>Figura 34.</b> Proceso para la creación y dibujo de patrones. ....	53
<b>Figura 35.</b> Histogramas HSV para el fondo vacío. ....	55
<b>Figura 36.</b> Histogramas HSV para el fondo con las fichas. ....	56
<b>Figura 37.</b> Ejemplo de segmentación por saturación; solo el rectángulo más saturado supera el umbral. ....	57
<b>Figura 38.</b> Procesamiento del segmentado; a) segmentación sin procesar, b) filtrado morfológico, c) detección de contornos, d) filtrado por tamaño de contorno. ....	58
<b>Figura 39.</b> Ejemplo ilustrativo (no real) de la extracción de características para un triángulo en un segmentado ya procesado. ....	59
<b>Figura 40.</b> Posiciones de una figura en la grilla del juego. ....	59
<b>Figura 41.</b> Árbol de decisión empleado para detectar la figura. ....	60
<b>Figura 42.</b> Ejemplo ilustrativo de la detección de objetos para un triángulo. ....	60
<b>Figura 43.</b> Calibración de la ubicación de la cámara para el tangram. ....	61
<b>Figura 44.</b> Plantillas empleadas para dibujar las siluetas del tangram. ....	62
<b>Figura 45.</b> Estructura básica del menú general. ....	63
<b>Figura 46.</b> Reconocimiento general de figuras. ....	64
<b>Figura 47.</b> Resultados de las etapas previas al reconocimiento; a) imagen sin procesar, b) filtrado de mediana, c) segmentación sin procesar, d) filtrado morfológico, e) detección de contornos, f) aproximación de los contornos a polígonos. ....	65
<b>Figura 48.</b> Reconocimiento de la flecha y su dirección. ....	67
<b>Figura 49.</b> Algunas escenas de la aplicación. ....	68
<b>Figura 50.</b> Desarrollo del juego en varias plataformas. ....	69
<b>Figura 51.</b> Actividades con colores. ....	70
<b>Figura 52.</b> Actividades con tamaños. ....	71
<b>Figura 53.</b> Actividades con Figuras. ....	72
<b>Figura 54.</b> Actividades con patrones. ....	73
<b>Figura 55.</b> Implementación del tangram. ....	74
<b>Figura 56.</b> Prueba de jugabilidad. ....	75
<b>Figura 57.</b> Tutorial del juego. ....	76
<b>Figura 58.</b> Público infantil del Hogar los Jilgueros. ....	79

## LISTA DE ANEXOS

	pág.
<b>Anexo A.</b> Histogramas HSV en respuesta a los colores de interés.....	85
<b>Anexo B.</b> Estructura del programa para el PDI. ....	88
<b>Anexo C.</b> La neuroeducación .....	93

## RESUMEN

El presente trabajo está dirigido a contribuir como un instrumento de apoyo en el aprendizaje de nociones básicas de la materia en un contexto geométrico, tales como el tamaño relativo, los colores, las figuras y algunos patrones simples, en niños mayores de 3 años. Lo mencionado, se puede alcanzar mediante el uso de tecnología aplicada a la visión por computador y el juego educativo, haciendo hincapié en el atractivo hacia lo tecnológico por parte de los infantes y promoviendo el aprendizaje divertido en apoyo a la falta de motivación de los pequeños ante el sistema de educación tradicional y el poco uso de la tecnología en el aspecto educativo, con la finalidad de expandir el proyecto en un futuro hacia áreas del conocimiento más complejas, como la matemática y la física, brindando un apoyo educativo y a la vez divertido.

*Palabras clave:* Patrones simples, Visión por Computadora, Juego Educativo.

## 1. INTRODUCCIÓN

Desde muy temprana edad los niños empiezan a aprender las nociones y conceptos básicos del mundo que los rodea, y es responsabilidad del adulto el instruir al niño en su aprendizaje, para ello existen una gama amplia de metodologías de enseñanza las cuales han ido evolucionando con el tiempo hasta la inmersión de la tecnología en el sistema educativo, a tal punto que en la actualidad no se puede ignorar la fuerte influencia de ésta en la educación de los infantes, en este contexto es importante dar un buen uso a la tecnología para estimular el aprendizaje de los niños. Por otra parte, es claro que los pequeños aman los juegos y rechazan las actividades de disciplina rígida según el modelo de enseñanza de Decroly (Ovide Decroly, 2002) y según el método Montessori *“La educación debería implementarse mediante lúdicas de juego para estimular el autoaprendizaje y curiosidad natural del niño y siguiendo el ritmo de aprendizaje del mismo”* (Palacios, 2018) . Teniendo en cuenta lo anterior y con base en la importancia de la tecnología y los juegos en la formación educativa de los infantes, se propone en este trabajo un método de apoyo a la enseñanza a partir del procesamiento digital de imágenes, con el cual se puedan usar lúdicas de juego que puedan contribuir al aprendizaje de conceptos básicos de la materia en un contexto geométrico, tales como formas, tamaños, colores y estimular el reconocimiento de algunos patrones simples.

## **2. PLANTEAMIENTO DEL PROBLEMA**

Es bien conocido que hoy en día existe un fuerte apego por parte de los niños hacia las nuevas tecnologías, de tal manera que se implementan e investigan nuevos métodos de enseñanza que las empleen puesto que son importantes para la motivación de los pequeños a aprender, pero el uso de las nuevas tecnologías por parte de los niños está enfocado en su mayoría al entretenimiento sin aprendizaje, esto tiene preocupados a muchos pedagogos infantiles, y hasta a instituciones como la Academia Americana de Pediatría y la Sociedad Canadiense de Pediatría (Pediatrics, 2013), en este ámbito es importante que haya un control en el uso de las nuevas tecnologías y generar un interés por el aprendizaje con las mismas tecnologías que tanto agrada a los pequeños. También es de vital importancia recalcar el rechazo de los niños en general hacía el sistema educativo tradicional, tal que ven el aprendizaje como algo malo y aburrido, lo que los lleva a estudiar sin ganas ni motivación y mucho menos diversión. En conclusión, el problema es la ausencia de motivación de los infantes ante el sistema de educación tradicional, y el desperdicio o poco uso de las nuevas tecnologías en el ámbito educativo por parte de los pequeños. En este contexto, se propone una contribución a la solución del problema con el sistema a implementar, debido a que se pretende que el niño (de cuatro años en adelante) pueda desarrollar habilidades a partir de la interacción con figuras que serían analizadas por un computador y capturadas por una webcam para que el niño aprenda, se divierta con lúdicas de juego e interactúe con el mundo físico que le rodea; todo esto bajo la guía de un instructor responsable.

### 3. JUSTIFICACIÓN

La formación de los pequeños es uno de los pilares fundamentales de la sociedad en que vivimos, puesto que de ello dependen las personas futuras en las que se convertirán los niños del presente, esto hace imperativo que la metodología empleada para educarlos deba ser agradable para ellos si fuere posible, para tal fin, en la actualidad se emplean lúdicas con juegos y aplicaciones en tecnología educativa, por ello se justifica la implementación del presente trabajo que pretende apoyar una pequeña parte del aprendizaje infantil y al mismo tiempo divertir e incluso se podría en un futuro establecer actividades en grupos para que el educando pueda interactuar y mejorar sus habilidades sociales, y así se lucharía contra el aislamiento social y la poca interacción del niño con su medio al usar excesivamente la tecnología moderna. En resumen, el presente trabajo busca brindar una alternativa divertida como apoyo al aprendizaje de características, tales como el color, la forma, el tamaño y el reconocimiento de patrones simples, a niños dentro de un rango de cuatro años en adelante (puesto que a partir de esta edad generalmente empiezan a reconocer puzles sencillos y a clasificar y ordenar objetos (Roldán, 2018)) con la esperanza de que pueda extenderse en un futuro para apoyar el aprendizaje de conceptos más complejos en matemáticas, geometría y física si fuere viable.

## **4. OBJETIVOS**

### **4.1. OBJETIVO GENERAL**

Elaborar un sistema de análisis y reconocimiento de figuras geométricas utilizando el procesamiento digital de imágenes, en conexión con un video juego, para elaborar lúdicas de juego que contribuyan en la percepción del infante referente a formas, tamaños, colores y el reconocimiento de patrones simples.

### **4.2. OBJETIVOS ESPECÍFICOS**

- Desarrollar un software capaz de reconocer y analizar figuras geométricas.
- Diseñar e implementar los juegos para las lúdicas.
- Crear una interfaz amigable con el público infantil en el rango establecido.



## 5. MARCO TEÓRICO

### 5.1. EL JUEGO EDUCATIVO

Durante este trabajo se ha planteado al juego educativo como la herramienta fundamental que impulsa el componente ingenieril en pro del aprendizaje de conceptos básicos referentes a la materia en un contexto geométrico plano (colores, tamaños, figuras y patrones), pero para entender el juego educativo, es necesario definir el concepto de *gamificación* o también conocido como *ludificación*, el cual cuenta con muchas definiciones, todas ellas bastante consistentes, pero aquí se ha tomado una recopilación general del término ya desarrollada en otro estudio en donde se define la gamificación como “*el uso de estrategias, modelos, dinámicas, mecánicas y elementos propios de los juegos en contextos ajenos a éstos, con el propósito de transmitir un mensaje o unos contenidos o de cambiar un comportamiento, a través de una experiencia lúdica que propicie la motivación, la implicación y la diversión*” (Faraón Llorens-Largo, 2016). En este contexto se destaca al juego educativo como un agente motivador que se fundamenta en la diversión para alcanzar los objetivos establecidos en el proceso de aprendizaje, esto se apoya en el señalamiento de la ciencia cognitiva de que aquello que el cerebro aprende es algo que capta la atención y genera emoción, de esta manera el juego educativo se convierte en un fuerte aliado del aprendizaje mediante el uso de la gamificación.

Cabe resaltar que la ludificación no solo se refiere a generar un ambiente atractivo para quien aprende como: bellas interfaces, colores, barras, puntuación, etc. Es necesario tener en cuenta los componentes esenciales de los juegos para así manipularlos y emplearlos en el contexto educativo y de esta manera hacer un buen desarrollo de aquello que se pretende transmitir o enseñar dentro del ambiente de juego. Estos componentes clave son: la diversión, la motivación, la autonomía, la progresividad, la retroalimentación inmediata y el tratamiento del error. Mediante la adecuada incorporación de estos aspectos es que se desarrolla la gamificación de manera eficiente siendo los mayores de ellos la motivación y la diversión; el cerebro aprende y como recompensa se divierte.

#### 5.1.1. Algunos teóricos relevantes del juego educativo

A continuación, se presentan algunos de los autores relevantes del juego educativo tomados de un artículo de la revista ‘*Educación*’ (Maureen Meneses Montero, 2001)

- **Groos** (1896), estudió el juego en los animales y posteriormente en los seres humanos; esto le permitió conocer las características innatas en las especies.
- **Claparède** (1900) continúa el estudio de Groos y define la teoría del pre ejercicio: ejercicios de tendencias instintivas que posteriormente encontrarían en la vida adulta, su aplicación concreta en el trabajo, en el deporte o en la vida cotidiana.

El juego, explica, es un ejercicio preparatorio para la vida y tiene como objeto el libre desarrollo de los instintos heredados todavía sin formar; resulta un agente natural educativo.

- **Wallon**: considera que el juego se confunde bastante bien con la actividad entera del niño, mientras esta siga siendo espontánea y no reciba sus objetivos de las disciplinas educativas. Por tanto, la escuela debe buscar en el juego infantil un medio y no condicionarlo a finalidades educativas cerradas.
- **Froebel** (1782-1852), pedagogo alemán, inicia el movimiento de educación preescolar sistemática; de él surgen los centros preescolares como necesidad social y familiar, método natural y activo ya que toma en cuenta la naturaleza infantil y su espontaneidad.
- **María Montessori**, (1870-1952) se basa en la teoría para el desarrollo y la liberación de los niños. Considera que el niño necesita mucho cariño pero que está dotado de una inmensa potencialidad latente, el niño es inquieto y en continua transformación corporal.

Los principios que fundamentan esta teoría en relación con el niño son: libertad, actividad, vitalidad, individualidad.

La escuela debe brindar al niño un ambiente apropiado en el que pueda actuar con total libertad y pueda encontrar el material y los juguetes didácticos que respondan a su profunda necesidad de moverse, actuar y realizar ejercicios.

- **Decroly** (1871-1932), trabajó para adecuar las leyes del desarrollo infantil a la enseñanza con el fin de atender las necesidades de los niños de manera integral.

Los principios fundamentales de su método son:

1. Principio de la globalización
2. Principio del interés

Basado en estos principios contribuye, por medio de la iniciación, a la actividad intelectual y motriz, a los juegos educativos.

Adopta la siguiente clasificación:

- a. Juegos que se refieren al desarrollo de las preocupaciones sensoriales y la aptitud motriz.
  - b. Juegos de iniciación matemática.
  - c. Juegos que se refieren a la noción del tiempo.
  - d. Juegos de iniciación a la lectura.
  - e. Juegos de gramática y comprensión del lenguaje.
- **Celestín Freinet** (1896-1966) propone la teoría de “la educación por el trabajo”. Parte de la búsqueda práctica de la educación popular interesante, eficiente y humana, sobre todo, en la cual el trabajo se constituye en eje y motor de su desarrollo.

Algunos de los principios que se deben considerar son: que se aprende por actividad específica, esto es, se aprende a leer y escribir leyendo y escribiendo; a dibujar se aprende dibujando, etc.; es decir, por medio de la libre exploración y la experimentación el niño aprende y conforma su inteligencia y sus conocimientos.

En este contexto es irrefutable el hecho de que el juego educativo es parte fundamental en el desarrollo cognoscitivo de los niños, que ha sido motivo de múltiples estudios y que sigue evolucionando. En el **Anexo C** se da un enfoque al juego educativo desde el punto de vista de la neuroeducación para dar ampliación a este tema desde los aportes de la neurociencia a la educación infantil.

### **5.1.2. El tangram y sus beneficios**

Un claro ejemplo de juego educativo es el tangram o tangrama que es un conocido juego de astucia de origen chino que se usa en el ámbito educativo por sus múltiples beneficios al desarrollo cognoscitivo de los niños. Según la Psicóloga y Psicopedagoga Laura Aguilera, el tangram es usado en psicología y pedagogía infantil, ya que promueve el desarrollo de las capacidades intelectuales y psicomotrices de los niños, permitiendo a su vez, enlazar la

manipulación concreta de materiales con la formación de ideas abstractas. Así, se puede hacer uso del tangram para desarrollar múltiples capacidades, las cuales, se citan a continuación. (Aguilera, 2015):

- Promueve el desarrollo de las capacidades psicomotrices e intelectuales.
- Facilita el aprendizaje de la geometría plana para niños
- Estimula la creatividad y contribuye a la formación de las ideas abstractas.
- Fomenta la orientación y estructuración espacial: aprender y relacionar unos objetos con otros en relación a la posición en la que están (arriba, abajo, izquierda y derecha).
- Desarrolla el conocimiento lógico-matemático: ayuda a realizar actividades relacionadas con ángulos, distancias, proporcionalidad, semejanza y movimientos.
- Permite entrenar la coordinación visomotora: coordinación ojo-mano.
- Mejora la atención: concentración y tiempos de atención sostenida para la realización de las figuras.
- Trabaja la percepción visual: interpretar y discriminar unos estímulos visuales de otros (diferentes figuras).
- Entrena la percepción de figura y fondo: permite diferenciar entre la figura y la composición de sus partes.

### **5.1.3. Las TIC en la educación temprana**

Ahora se prosigue a presentar un enfoque más actual, donde las tecnologías de la información y la comunicación (TIC) juegan un rol importante en la educación infantil.

Las TIC son la realidad del día a día de las personas de este siglo, ya sea a través de smartphones, Tablets, laptops, smartTVs, consolas de videojuegos, etc. La humanidad moderna está ligada a estos dispositivos de tal forma que es inevitable omitir la importancia de su inmersión en el sistema educativo.

La especialista Fabiola Ruiz analiza las TIC en la educación inicial: *“Pensar el trabajo con las TIC en la Educación Inicial es posible si las comprendemos y las incluimos como un eje transversal en la tarea cotidiana de enseñanza. Es decir, como un conjunto de herramientas que —acompañadas de diversas estrategias*

*docentes— estén incluidas en la planificación de proyectos o recortes del ambiente a fin de potenciar nuevos y/o mejores aprendizajes”.* (Fabiola, 2015). En este punto es claro que las TIC se convierten en herramientas para fomentar el aprendizaje, pero teniendo en cuenta que se deben usar bajo estrategias adecuadas para no desviarse de los objetivos del entorno educativo.

Las TIC son usadas muy frecuentemente en los jardines o en los grados inferiores, donde el ambiente académico es menos rígido y es más difícil captar la atención de los infantes. En estos espacios es común usar Tablets o computadores para mostrar imágenes previamente seleccionadas por los docentes para cumplir con objetivos referentes al ámbito académico, también se usan videos, cancioneros, etc. La finalidad de estas técnicas es lograr captar la atención del niño sin la necesidad de acudir a la disciplina rígida, sino que se trata de redireccionar el interés innato del pequeño por todo aquello que está relacionado con la tecnología y convertirlo en un impulso motivador por el aprendizaje.

#### **5.1.4. La visión por computador y juego educativo**

Dado el avance y el gran alcance de las aplicaciones que se pueden desarrollar con el procesamiento digital de imágenes, el ámbito educativo no podía quedar excluido. Un ejemplo bastante notorio de esto es el software de *“realidad aumentada”* del Ingeniero en Sistemas y Magister en Psicología Educativa Juan Raúl Cadillo León, quien es un docente peruano ganador del premio “El Maestro que Deja Huella” en el año 2014 y nominado al premio “Global Teacher Prize” (Cadillo León, 2012). El software del profesor Cadillo León, se basa en reconocer figuras básicas del ambiente mediante una webcam y generar animaciones en los mismos *“frames”* de la cámara, con el propósito de enseñar mediante el juego, conceptos matemáticos simples, especialmente a niños pequeños como lo hace en su aplicación *“Subsistema de manejo de los Bloques Lógicos”* (ver **Figura 1**), en el cual se deben ordenar los elementos que aparecen en la cámara por su color tamaño y forma.



*Figura 1. Subsistema de manejo de los Bloques Lógicos (Cadillo León, 2012).*

Después de este breve contexto educativo, se procede a presentar los factores más relevantes en términos más ingenieriles en lo que se refiere al desarrollo técnico de este trabajo, en especial al componente de visión por computador que es el centro de enfoque desde la perspectiva ingenieril.

## **5.2. PROCESAMIENTO DIGITAL DE IMÁGENES**

El procesamiento digital de imágenes (PDI), se refiere a un conjunto de técnicas o procesos para descubrir o resaltar información contenida en una imagen digital, usando como herramienta principal un computador (Torres, 1996). Esta tecnología se fundamenta en el álgebra matricial, ya que para un computador una imagen es como una matriz de píxeles en donde se encuentra la información digital referente a un espectro de radiación, como el espectro visible o alguna de las bandas del infrarrojo, como se suele usar en algunas imágenes satelitales.

### **5.2.1. Aplicaciones**

Las aplicaciones son muy diversas y actualmente son de gran utilidad, como lo son la percepción remota; referente a las imágenes tomadas desde el espacio, la detección de rostros, que es usada en los sistemas de vigilancia y el análisis de imágenes médicas con énfasis en de la detección de anomalías que podrían deteriorar la salud entre otras aplicaciones. En este contexto se entiende que el PDI es una gran herramienta en esta era de información digital y que puede tener uso en otras áreas como en la educación, aprovechando el fácil acceso a los computadores en la mayor parte del mundo.

### 5.2.2. Características básicas de una imagen digital

Una imagen digital es información cuantizada y finita almacenada de manera similar a una matriz, en la cual cada posición está representada por un pixel que contiene la información de la radiación, o particularmente en este caso del color, a lo cual se le denomina niveles de gris como se ilustra en la **Figura 2**.

255	255	254	255	255	255	255	254	255	255
255	244	254	255	255	255	255	254	244	255
236	223	254	255	255	255	255	254	225	234
184	200	232	247	252	252	247	233	202	182
131	163	190	206	214	214	207	191	164	128
103	116	142	157	165	165	158	143	117	99
134	78	91	104	111	111	105	92	78	129
214	76	75	75	76	76	75	75	76	210
255	177	76	75	75	75	75	76	172	255
255	255	214	133	95	95	132	211	255	255

*Figura 2. Imagen digital.*

Entre las características básicas de una imagen digital, se destacan las siguientes:

- **Resolución:** se refiere a la cantidad de pixeles que contiene la imagen y se denota como las dimensiones de una matriz. En el caso de la **Figura 2**, se tiene una imagen de 10x10 pixeles (en un modo simple se refiere a la grilla)
- **Profundidad del pixel o niveles de gris:** es el rango de valores que puede tomar el pixel, siendo el valor máximo el blanco y el valor mínimo el negro, en el ejemplo de la **Figura 2** que es el caso típico para un trabajo como este, el valor mínimo es cero y el máximo es 255, esto se debe a que la información digital es almacenada en bits que pueden tomar el valor de cero o uno, y en este caso las imágenes son de ocho bits en principio, aunque es posible obtener imágenes de solo dos niveles de gris, a las cuales se les denomina imágenes binarias (se obtienen en el proceso de segmentación).
- **Relación de aspecto:** se refiere a la razón entre el ancho y alto de la imagen (X: Y)

- **Planos de color o canales:** son equivalentes a la dimensión de profundidad de una matriz; las imágenes a blanco y negro solo tienen un plano, mientras que las imágenes a color tienen tres planos o más, y en cada plano se guarda información que se complementa para obtener el color del pixel si se trabaja en el espectro visible, o en general el nivel de gris.
- **Modelo de color:** éste determina cómo se almacena la información en los planos de color, por ejemplo, el modelo RGB (rojo, verde y azul por sus siglas en inglés), guarda la información del color rojo en el primer plano, del verde en el segundo y del azul en el tercero. Existen varios modelos de color aparte del RGB que es el más común, como el modelo HSV, el cual representa el color, la saturación y el brillo respectivamente, que es muy útil para identificar el color independientemente de la saturación y el brillo tomando únicamente el plano H.

### 5.2.3. Etapas del procesamiento

El PDI debe someterse a varias fases las cuales pueden variar según el tipo de aplicación, pero tomando un estándar de procesos se pueden ordenar las etapas del proceso de la siguiente manera (Wainschenker, 2011):

- **Captura:** esta parte hace referencia al diseño de las propiedades de la captura, como el tipo cámara, la resolución, la distancia al objeto, la iluminación, etc.
- **Preprocesamiento:** en esta sección se pretende reducir el entorno que no es de interés para el problema. Fondo, ruido, etc.
- **Segmentación:** aquí se trata de reconocer y extraer cada uno de los objetos presentes en la imagen que sean de interés.
- **Extracción de características:** se refiere a seleccionar y extraer “características” apropiadas para la identificación de los objetos deseados.
- **Detección de objetos:** Esta es la parte final, consiste en utilizar un modelo de toma de decisión para decidir a qué categoría pertenece cada objeto.



## 5.3. RECONOCIMIENTO DE FIGURAS GEOMÉTRICAS

### 5.3.1. OpenCV

En el ámbito del reconocimiento de imágenes existe una herramienta muy desarrollada llamada OpenCV, la cual es un conjunto de librerías utilizadas en este trabajo con el fin de llevar a cabo la detección de figuras.

OpenCV es una biblioteca de visión artificial de código disponible en <https://sourceforge.net/projects/opencvlibrary/>. La biblioteca está escrita en C y C ++ y se ejecuta en Linux, Windows y Mac OS X. Hay un desarrollo activo en interfaces para Python, Ruby, Matlab y otros lenguajes.

OpenCV fue diseñado para la eficiencia computacional y con un fuerte enfoque en aplicaciones en tiempo real. OpenCV está escrito y optimizado en C y puede aprovechar los procesadores multinúcleo. Uno de los objetivos de OpenCV es proporcionar una infraestructura de visión por computador fácil de usar, eso ayuda a las personas a construir aplicaciones de visión bastante sofisticadas rápidamente. La biblioteca contiene más de 500 funciones que abarcan muchas áreas de la visión, incluidas la inspección de productos en fábricas, imágenes médicas, seguridad, interfaces de usuario, calibración de cámaras y la robótica (Gary Bradski, 2008).

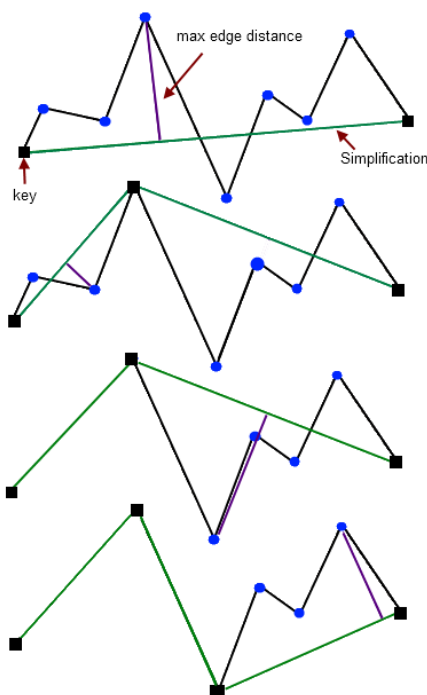
### 5.3.2. Algoritmo de Ramer–Douglas–Peucker

Una vez que se ha segmentado y filtrado la imagen, entonces se tendrá una imagen binaria (solo dos valores en los niveles de gris, el blanco y el negro), la cual contiene los contornos asociados a las figuras geométricas (ver **Figura 3**), se procede a implementar una útil herramienta contenida en las librerías de OpenCV llamada “approxPolyDP” la cual es una función que emplea el algoritmo de Ramer–Douglas–Peucker, también conocido como el algoritmo de Douglas–Peucker.



**Figura 3.** Imagen binaria preparada para detectar los contornos o figuras geométricas.

El algoritmo de Douglas–Peucker consiste en reducir una curva compuesta por varios segmentos de línea, a la menor cantidad de puntos o segmentos posibles dentro de un margen de error específico (épsilon), esto se logra bajo el proceso ilustrado en la **Figura 4**.

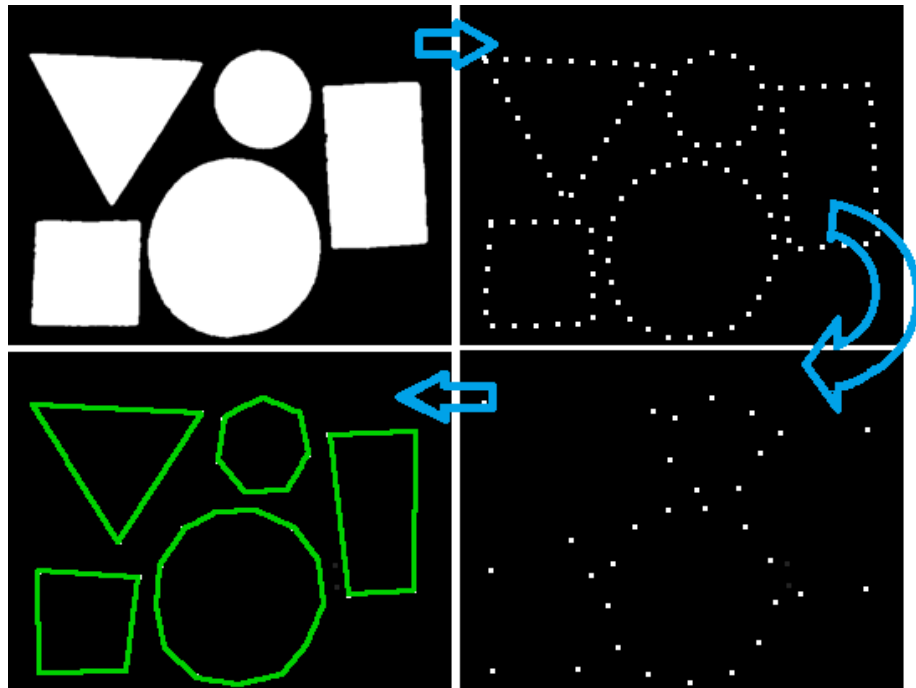


**Figura 4.** Implementación del algoritmo de Douglas-Peucker. (Koning, 2011).

La primera fase consiste en tomar una recta a través de la línea poligonal desde el primer punto hasta el último, luego se ubica el punto con la mayor distancia perpendicular a esta recta, si la distancia perpendicular es mayor a épsilon todos los puntos intermedios serán suprimidos, pero si no lo es, entonces se toma este último punto, como referente para medir la mayor distancia perpendicular entre los puntos intermedios entre este y el primero y también con el último, este proceso se hace reiterativamente como se ve en la figura y causará la eliminación de los puntos cuyas distancias perpendiculares sean menor al

épsilon, generando así una nueva curva o polilínea que contiene una cantidad menor de segmentos.

La función de OpenCV que separa los contornos en la **Figura 3** (findContours) entrega como resultado un polígono de muchos puntos por cada contorno (ver **Figura 5**); en este punto se usa la función “approxPolyDP” para reducir la cantidad de puntos a la misma cantidad de vértices de la figura geométrica más similar.



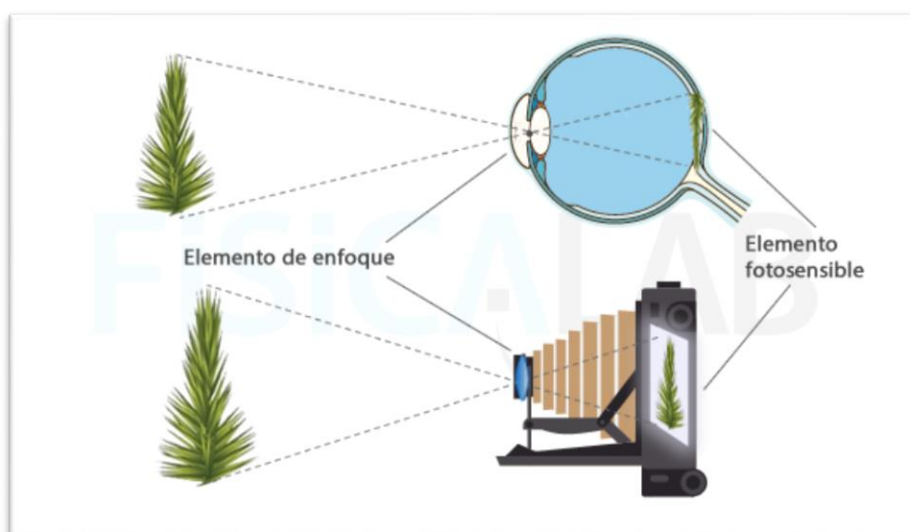
*Figura 5. Ejemplo de aplicación del algoritmo de Douglas-Peucker.*

Finalmente, la función entrega un arreglo que contiene los puntos reducidos de cada contorno, por ejemplo, si los puntos reducidos de un contorno fueron tres, esto quiere decir que lo más probable es que ese contorno sea un triángulo. Cabe aclarar que la precisión de la detección depende del épsilon que se emplee; si se usa un épsilon pequeño no se tendrá inconvenientes con los círculos, pero las figuras que son más rectangulares pueden aproximarse con más puntos, lo que dificultaría su detección, y un épsilon grande aproximaría un círculo con un polígono de muchos lados, aunque las otras figuras se aproximarían mejor. En este contexto es importante encontrar un valor óptimo para épsilon dentro de las necesidades de la aplicación.

#### 5.4. CARACTERÍSTICAS DE UNA CÁMARA DIGITAL

Con el propósito de obtener la mejor configuración para la cámara web que se usa en este trabajo, se realiza un enfoque teórico al funcionamiento de una cámara digital ya que una webcam es simplemente una cámara digital optimizada para un funcionamiento en conexión directa con un computador, generalmente con fines relacionados a la conexión web.

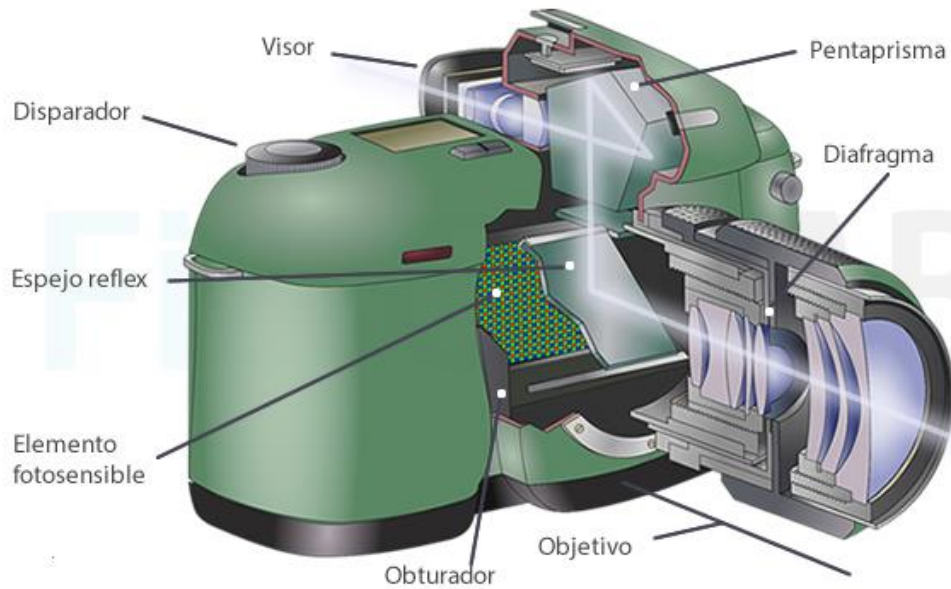
El principio físico de una cámara es en esencia el mismo que rige el comportamiento del ojo humano (ver **Figura 6**). La luz proveniente de un objeto imagen llega a una lente biconvexa y es reunida en un punto espacial llamado foco, el cual define la distancia a la que se forma la imagen (distancia focal), en este punto la luz impacta en un área de un elemento fotosensible (como la retina en el ojo humano), aunque en el caso de las cámaras digitales, lo que es sensible a la luz es un sensor de imagen que convierte la información lumínica en señales eléctricas que son digitalizadas. La imagen obtenida está generalmente al revés, pero al igual que el cerebro humano hace la corrección mediante una transformación similar a la que se le hace a una matriz para girarla, las cámaras también hacen un preprocesamiento para obtener la imagen corregida, aunque también se suelen usar lentes para corregir la imagen. Generalmente un dispositivo de esta clase es un poco más complejo, ya que cuenta con varias lentes de corrección y un sistema llamado visor para cambiar las distancias a las que se puede enfocar una imagen según los requerimientos de la fotografía. De esta manera se obtiene una representación en 2D de un objeto en 3D, que es digitalizada para poder ser procesada mediante transformaciones matriciales en un computador.



**Figura 6.** Principio físico de una cámara (Fernández, 2013).

### 5.4.1. Elementos de una cámara

Una cámara consta de varias partes físicas que definen la imagen que se obtendrá como se ve a continuación.



**Figura 7.** Componentes de una cámara (Fernández, 2013).

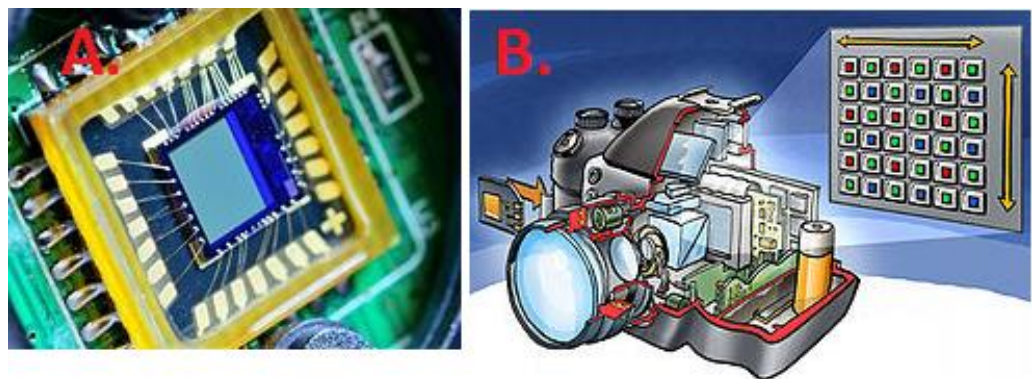
En la **Figura 7** se observa una cámara general; en este enfoque se ignora los componentes que no posee una cámara web de gama media como el disparador, el visor, el pentaprisma y el espejo réflex, con esta aclaración, las partes que son de interés en este contexto son:

- **El objetivo:** como se muestra en la imagen anterior, el objetivo es un sistema de lentes que, en conjunto deben comportarse como una lente convergente ideal. Su función es formar la imagen real e invertida sobre el fondo de la cámara (Fernández, 2013)
- **El diafragma:** éste es un sistema dentro del visor que controla la cantidad de luz que ingresa al sensor, esto se logra controlando el diámetro de una apertura que limita el paso de luz como se ve en la **Figura 8**.



**Figura 8.** Diafragma de una cámara (Fernández, 2013).

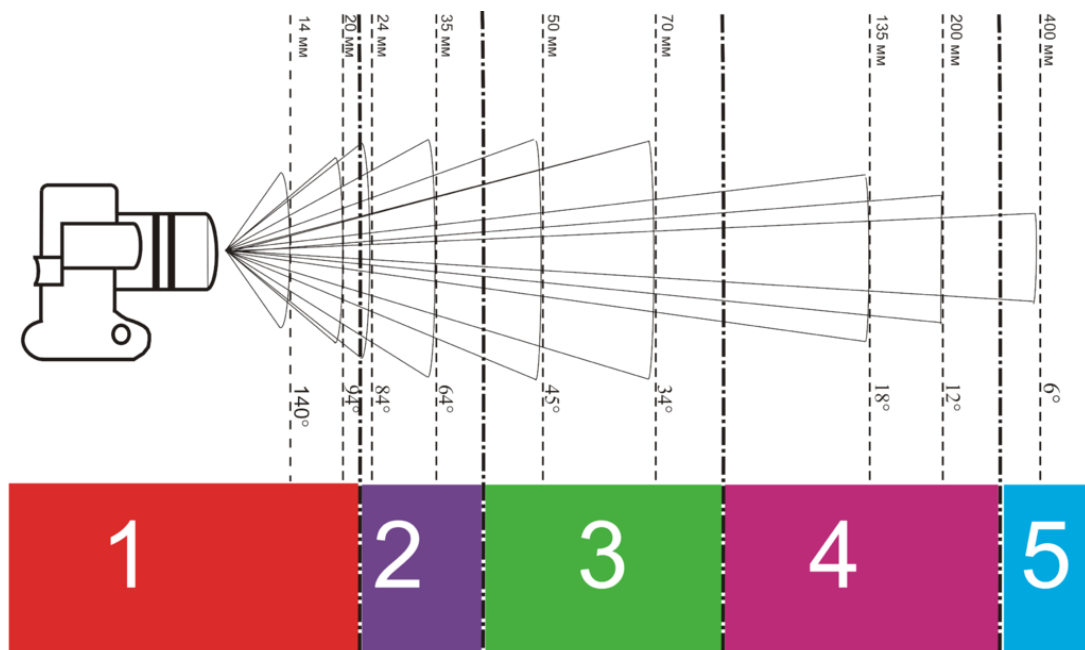
- **El obturador:** éste es un mecanismo a modo de cortinilla que se abre para que la luz impacte en el sensor durante un tiempo determinado (tiempo de exposición), este tiempo define para una webcam la velocidad de muestreo de los fotogramas, esto quiere decir que entre mayor sea el tiempo de exposición, mayor será la cantidad de luz para formar la imagen, pero a la vez esto implica que la imagen tardará más en crearse, lo que implica un baja velocidad de fotogramas por segundo (*framerate*).
- **El elemento fotosensible:** es simplemente el encargado de formar la imagen. En la cámara digital es un sensor de imagen que separa la luz incidente en sus componentes rojo verde y azul, para luego generar a partir de esto impulsos eléctricos que son discretizados y convertidos en información digital como se ve en la **Figura 9-B**.



**Figura 9.** **A.** Sensor de imagen de una cámara web. **B.** conversión de la imagen en digital. (Contreras, 2013) (Welleman, 2008).

### 5.4.2. Parámetros a tener en cuenta

- **Resolución:** se refiere a la cantidad de píxeles que contiene la imagen generada por el dispositivo. Por ejemplo, en este caso en particular se cuenta con una cámara web estándar de 640x480p
- **Frame-rate o FPS (frames per second):** es un parámetro especialmente para las webcams y se refiere a la velocidad máxima a la que se puede capturar fotogramas en una toma de video.
- **Exposición:** es el tiempo mínimo necesario durante el cual el obturador permanece abierto para tomar una fotografía (está relacionado con los FPS).
- **Ángulo de visión:** esta es una variable que determina o mide el campo de visión o la parte de la escena que es captada en el sensor como se ve en la **Figura 10**.



**Figura 10.** Ángulo de visión según distancia focal del objetivo de una cámara. Diagrama sobre la relación entre la distancia focal del objetivo y el ángulo de visión abarcado: 1. Objetivo ojo de pez. 2. Objetivo gran angular. 3. Objetivo normal. 4. Teleobjetivo. 5. Súper teleobjetivo (MrSpyDoS, 2012).



## 6. PROCEDIMIENTO EXPERIMENTAL

### OBJETIVOS ESPECÍFICOS

- ✓ Diseñar e implementar los juegos para las lúdicas
- ✓ Desarrollar un software capaz de reconocer y analizar figuras geométricas.
- ✓ Crear una interfaz amigable con el público infantil en el rango establecido

En la búsqueda del cumplimiento de los objetivos específicos se ha planteado un sistema de tres componentes fundamentales: el escenario donde se llevan a cabo las actividades (entorno físico), las actividades lúdicas para transmitir el conocimiento deseado y la aplicación a modo de videojuego donde todo se interrelaciona (ver **Figura 11**).



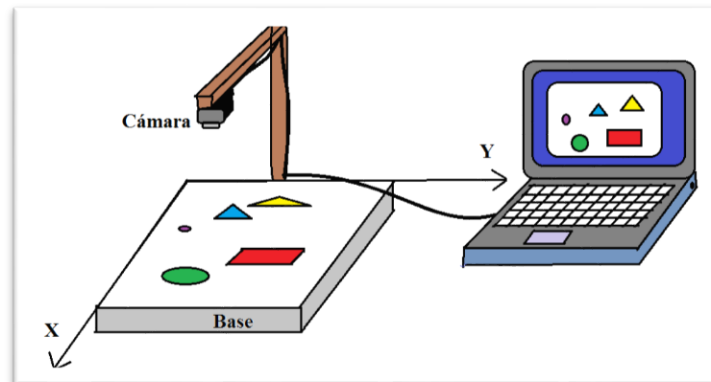
**Figura 11.** Arquitectura general para el desarrollo de la herramienta.

### 6.1. ENTORNO FÍSICO

El montaje físico es la primera etapa del proceso para alcanzar el cumplimiento de los objetivos, este define el escenario para las actividades, a la vez que impone sus limitantes. En este apartado se cuenta con un conjunto de figuras con las que se controlará la aplicación a modo de mando y con las cuales se realizan las actividades, también se tiene una cámara o webcam para captar y luego procesar lo que sucede con las figuras y enviar esa información a la aplicación y finalmente se necesita un fondo o “background” como base para las



figuras o fichas y también para facilitar el reconocimiento de las figuras por medio del procesamiento digital de imágenes, así como se muestra en la **Figura 12**.



**Figura 12.** Esquema para el escenario físico.

Con el anterior esquema definido se procede a establecer las características particulares para cada uno de sus componentes principales.

### 6.1.1. Configuración de la cámara

Como se observa en la **Figura 12**, la cámara se ubica perpendicularmente a la base o fondo; la altura a la que se usará la cámara depende de sus características, especialmente de su ángulo de visión, dado que un valor pequeño para este parámetro implicaría que la cámara debe ubicarse mucho más alto para lograr captar todo el escenario de interacción que es de interés, y al alejarse del objetivo, éste puede perder calidad en la imagen, pero al contrario, si la cámara tiene un ángulo de visión amplio para un objetivo “gran angular o normal” (ver **Figura 10**), entonces la altura para la base de la cámara no necesita ser muy grande (no mayor a 52cm). La lente de la cámara en este caso en particular debe enfocarse a 52cm del objetivo y debe ser perpendicular al plano horizontal.

Para este trabajo se ha empleado una webcam estándar de gama media marca “Carrefour” modelo CWC21 (ver **Figura 13**) con sus correspondientes características mostradas en la **Tabla 1**.



**Figura 13.** Cámara usada durante la experimentación.

**Tabla 1**

*Especificaciones para la cámara utilizada en este trabajo.*

<b>Característica</b>	<b>Valor</b>
Frame-rate (FPS) máximo	30
Mínima distancia a imagen (cm)	5
Tipo de sensor	CMOS
Profundidad de color (bits)	8
USB	2.0
Resolución (píxeles)	640x480
Relación de aspecto	4:3
Distancia focal (mm)	38.5
Angulo de visión a 52 cm del objetivo	40°
Formato de captura	BMP/JPG

### **6.1.2. Iluminación**

Es importante para cualquier aplicación relacionada a la visión por computador, tener en cuenta la iluminación, principalmente cuando se trabaja con cámaras de bajo presupuesto, esto se debe a que estos dispositivos generalmente no permiten configurar la mayoría de sus características, por ejemplo, la velocidad de obturación que afecta directamente la tasa de fotogramas por segundo (FPS) o el tiempo de exposición, que en cámaras web como la usada en este trabajo son parámetros configurados únicamente por el software del dispositivo, el cual los varía según las condiciones de luz para obtener el mejor fotograma ya que el sistema de la cámara tratará de compensar la falta de luz con un mayor tiempo de exposición, lo que afectará los FPS para la adquisición de imágenes, esto podría generar una caída en el rendimiento del programa al generar un retraso en el bucle que hace la captura de frames en el código de la aplicación.

Con lo anterior en mente, se recomienda una iluminación controlada (suave, blanca y perpendicular en lo posible), puesto que una luz de colores afectaría la

percepción de color de la cámara y una iluminación dura y direccional generaría demasiados contrastes y sombras que dificultarían el proceso de segmentación.

### 6.1.3. Características para el fondo

Es primordial que el fondo sea claramente diferenciable de las fichas para disminuir carga en el proceso de segmentado. En este trabajo se ha decidido por comodidad y bajo costo, usar cartón paja u otro cartón de color uniforme y bajo en saturación como el beige claro o gris, lo que hace que las figuras resalten respecto al fondo por el contraste de saturación.

Las dimensiones para la base o fondo, deben ser superiores al campo de visión de la cámara (40x30 cm<sup>2</sup>), se recomienda que sean al menos 45x35 cm<sup>2</sup>.

### 6.1.4. Características para las figuras

Las fichas deben ser notables respecto al fondo, lo cual implica que deben pintarse con colores vivos o muy saturados, por lo cual solo se recomienda el uso de colores básicos: rojo, verde, azul, amarillo, cian y violeta. Las figuras pueden ser hechas en cartón paja, siempre y cuando queden bien coloreadas, se deben pintar por ambos lados para evitar que se curven al secar. No se debe usar pintura tóxica ya que las figuras serán manipuladas por niños, tampoco pueden ser demasiado pequeñas para que no sean ingeridas por accidente.

Con la finalidad de desarrollar las lúdicas, se han establecido tres grupos de figuras para las actividades con los niños pequeños (4 a 6 años) y un grupo adicional para niños más grandes (6 años en adelante). Las características de las fichas de cada grupo se muestran en las tablas siguientes.

**Tabla 2**

*Primer grupo de figuras.*

<b>Figura</b>	<b>Dimensiones (cm)</b>	<b>Color (cara1/cara2)</b>
Círculo	Radio=3	Rojo/Morado
Círculo	Radio=3	Verde/Amarillo
Círculo	Radio=3	Azul/Cian

*Se diferencian solo por el color.*

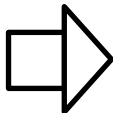
**Tabla 3***Segundo grupo de figuras.*

<b>Figura</b>	<b>Dimensiones (cm)</b>	<b>Color (ambas caras)</b>
Cuadrado	Lado=4	Azul
Cuadrado	Lado=6	Azul
Cuadrado	Lado=8	Azul

*Se diferencian solo por el tamaño.***Tabla 4***Tercer grupo de figuras.*

<b>Figura</b>	<b>Dimensiones (cm)</b>	<b>Color (ambas caras)</b>
Círculo 1	Radio=3	Amarillo
Triángulo equilátero	Lado=7	Amarillo
Cuadrado	Lado=6	Amarillo
Rectángulo	Ancho=6 / Alto=4	Amarillo
Pentágono	Inscrito en círculo de diámetro = 7	Amarillo
Hexágono	Inscrito en círculo de diámetro = 7	Amarillo

*Se diferencian por la forma geométrica.***Tabla 5***Cuarto grupo de figuras*

<b>Figura</b>	<b>Dimensiones (cm)</b>	<b>Color (ambas caras)</b>
Cuadrado de tangram	Lado=10	Libre
Círculo	Radio=7.5	Igual al fondo
Flecha 	Cuadrado de lado=4 Triángulo isósceles de base = 8 y altura =4	libre

*Son las figuras correspondientes al juego de astucia llamado “tangram” y una flecha que gira en un círculo para manejar el componente de direccionalidad.*

**Nota:** para la construcción de las figuras se ha creado una plantilla para imprimir o para servir de guía (disponible en: <http://bit.ly/2GP0yXn>) si se usa la plantilla a color, entonces se debe calibrar el color de la impresora para que los colores resulten como se ven en pantalla.

## 6.2. DESARROLLO DE LAS ACTIVIDADES LÚDICAS

✓ Diseñar e implementar los juegos para las lúdicas

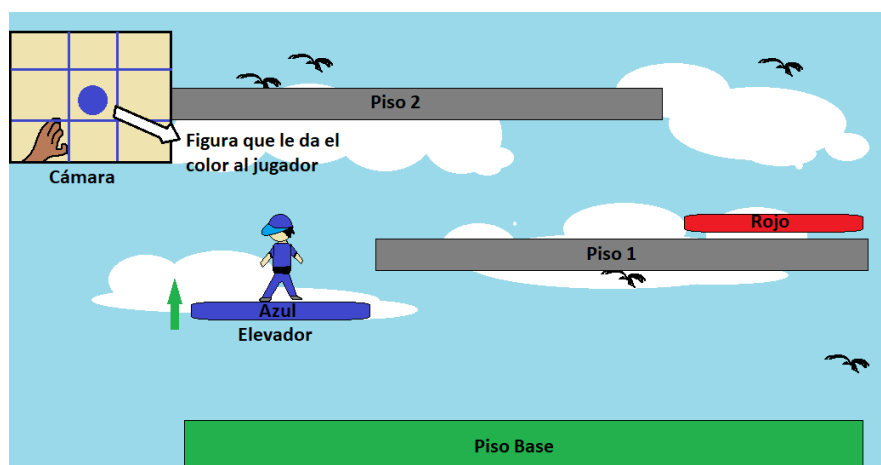
Una vez que se ha establecido el entorno físico se procede como segunda etapa a desarrollar las actividades o lúdicas que se van a realizar con los niños, éstas se hacen en forma de croquis para ser luego programadas en la siguiente etapa.

Luego de investigar los métodos más comunes mediante los cuales se enseña a percibir las características de interés en este trabajo, se rescata que se usan principalmente fichas de colores vistosos, que se diferencian solo por la variable de interés, por ejemplo: para identificar colores se utilizan figuras que únicamente se diferencian por el color. Dada la aclaración, se entiende el por qué se crean las fichas de la forma en la que se plantea en el apartado anterior.

### 6.2.1. Lúdicas para identificar colores

En esta parte se emplean las figuras de la **Tabla 2**, que son círculos diferenciados solo por el color; se han tomado tres fichas y seis colores, los cuales están a una misma distancia en el modelo de color HSV.

La actividad consiste en un personaje que debe subir una serie de pisos cada vez más numerosos, para lo cual debe usar un elevador que debe ser activado mediante el color, este color estará en el elevador y debe ser el mismo color que el personaje lleva en su atuendo, por lo cual se debe cambiar el color del jugador colocando en la cámara, la figura del grupo tres con el color correspondiente (ver **Figura 14**).



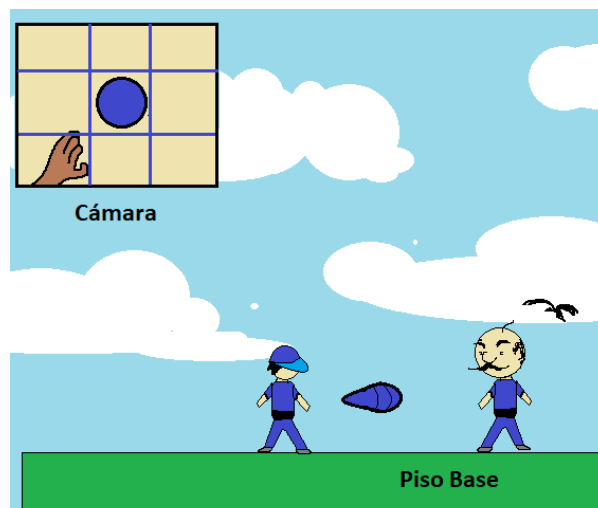
**Figura 14.** Esquema de juego con colores.

Una actividad adicional, consiste en ordenar las fichas por su color según el patrón de color que presentan un conjunto de mariposas animadas, para que de esta forma sean atrapadas por la red del color correspondiente para después ser liberadas con el fin de no incentivar practicas antiecológicas. Para entender mejor la lúdica ver **Figura 15**.



**Figura 15.** Segundo esquema de juego con colores.

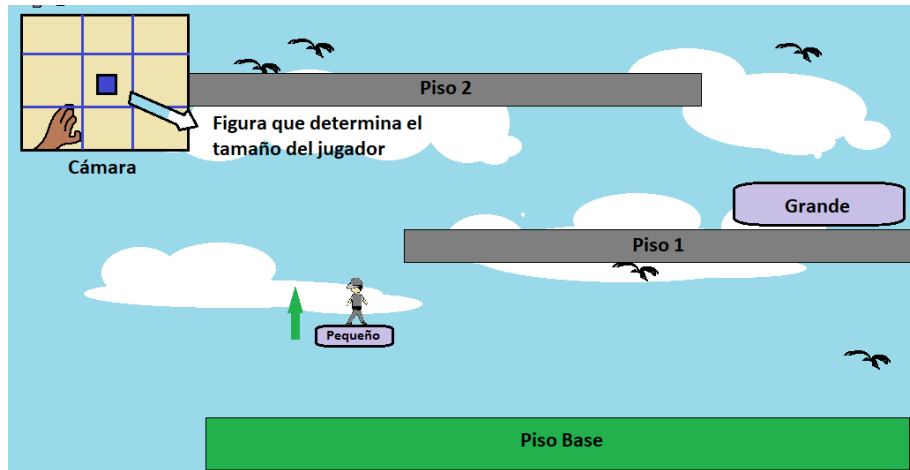
También se ha desarrollado una tercera actividad para agregar un poco de dificultad o incrementar el reto, el juego consiste en “congelar” a un rival que se opone a dejar pasar al jugador, como se ve en la **Figura 16** y para lograr superar este reto, el jugador debe lanzar una bala paralizante que solo tendrá efecto si lleva el mismo color que el enemigo en cuestión. El color del personaje implicará el color de la bala que se ajusta según sea la figura que está en la cámara.



**Figura 16.** Tercer esquema de juego con colores.

### 6.2.2. Lúdicas para identificar tamaños

En este apartado se trabaja similarmente al anterior, con la diferencia de que se usa el segundo grupo de fichas (ver **Tabla 3**), que se diferencian solo por su tamaño relativo (mediano, grande y pequeño). En la **Figura 17** se puede observar que el elevador tiene un tamaño asociado que debe coincidir con el tamaño del jugador para que se active, estos tamaños se consiguen colocando en la cámara la ficha con el tamaño relativo correspondiente.



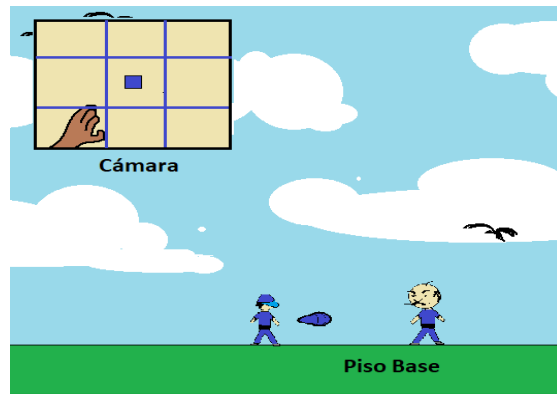
**Figura 17.** Esquema de juego con tamaños relativos.

Similarmente a la segunda actividad anterior, se trabaja otra lúdica que consiste en ordenar tamaños en la cámara según el tamaño de las mariposas, así como se muestra en la **Figura 18**. El orden de los tamaños relativos de las fichas en la grilla de la cámara determinará el orden de las redes para que así puedan atrapar a las mariposas; éstas solo entrarán en la red si su tamaño relativo es el correspondiente.



**Figura 18.** Segundo esquema de juego con tamaños.

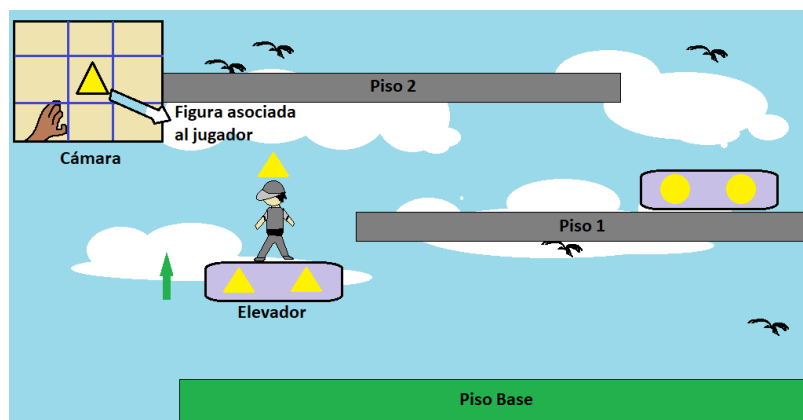
De la misma manera en la que se ha venido trabajando, se desarrolla una actividad similar a la anterior, pero con el tamaño relativo como variable en este caso, ahora las balas cambian de tamaño para tener efecto sobre el rival que también podrá ser de varios tamaños. En la **Figura 19** se muestra un ejemplo en el que el rival es pequeño, y por lo tanto el jugador debe hacerse pequeño para que sus proyectiles puedan congelar a su némesis, esto se logra activando el tamaño pequeño con la ficha correcta en la cámara.



**Figura 19.** Tercer esquema de juego con tamaños.

### 6.2.3. Lúdica para identificar figuras

Para promover la identificación de algunas figuras geométricas básicas se emplean las fichas de la **Tabla 4** que varían su forma geométrica. La actividad es similar a las anteriores, pero dado que el personaje carece de una propiedad que varíe según la forma geométrica, entonces se le agrega una como ítem de juego tal como se muestra en la **Figura 20**, y claramente se hace lo mismo con el elevador.



**Figura 20.** Esquema de juego con figuras.



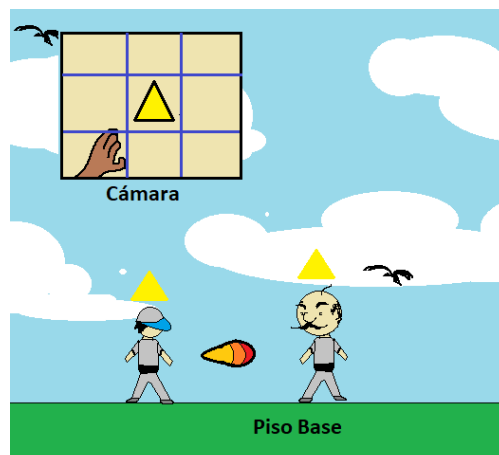
Como segunda actividad de juego se asocian a las mariposas figuras que dejarán caer en determinados botes que tienen sus fichas asociadas según las fichas que aparecen en la cámara como lo muestra la **Figura 21**. El juego consiste en ordenar las figuras en la cámara en el mismo orden en que las mariposas traen las figuras para que las fichas caigan en sus botes correspondientes. Si una ficha entra en un bote que no le corresponde, ésta rebotará y se perderá sin generar ninguna clase de puntuación.



**Figura 21.** Segundo esquema de juego con figuras.

Como tercera actividad se le asocia al rival del personaje una figura geométrica como se muestra en la **Figura 22** y para que las balas paralizantes tengan efecto, el jugador deberá tener la misma figura geométrica asociada.

**Nota:** Se destaca que los proyectiles solo pausan momentáneamente la animación o las acciones del rival del jugador al igual que en los apartados anteriores para que el protagonista pueda avanzar en su aventura. No se pretende promover ningún tipo de violencia en este juego.



**Figura 22.** Tercer esquema de juego con figuras.

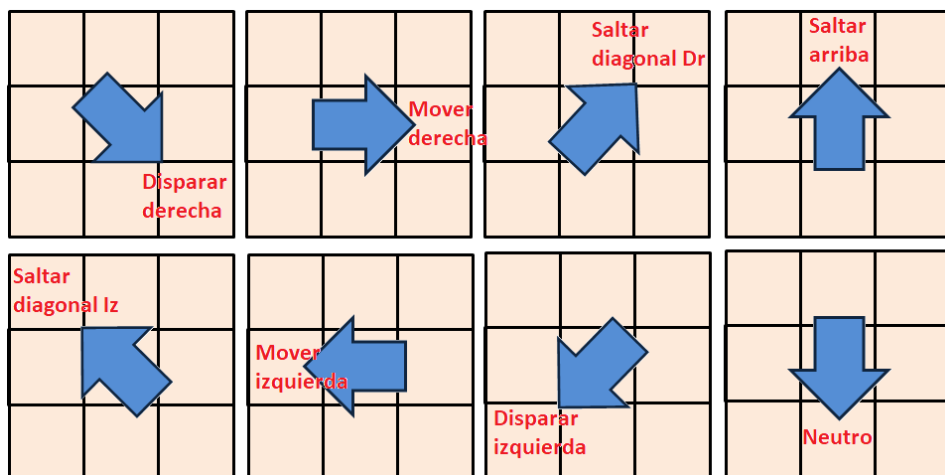
### 6.2.4. Componente de direccionalidad

No es objetivo de este trabajo, trabajar con la direccionalidad en los infantes, pero se ha agregado un aspecto de ella como componente adicional. Aprovechando que el personaje necesita moverse, en lugar de crear un movimiento automático, se ha tomado la decisión de asociar el movimiento a una flecha que gira alrededor de un eje que pasa por su centro geométrico como se ve en la **Figura 23**.



**Figura 23.** Esquema para el componente direccional del juego.

La dirección de la flecha indica la dirección de movimiento del jugador, pero dado que la figura puede rotar 360 grados, entonces se le asocian más acciones como se muestra en la **Figura 24**.

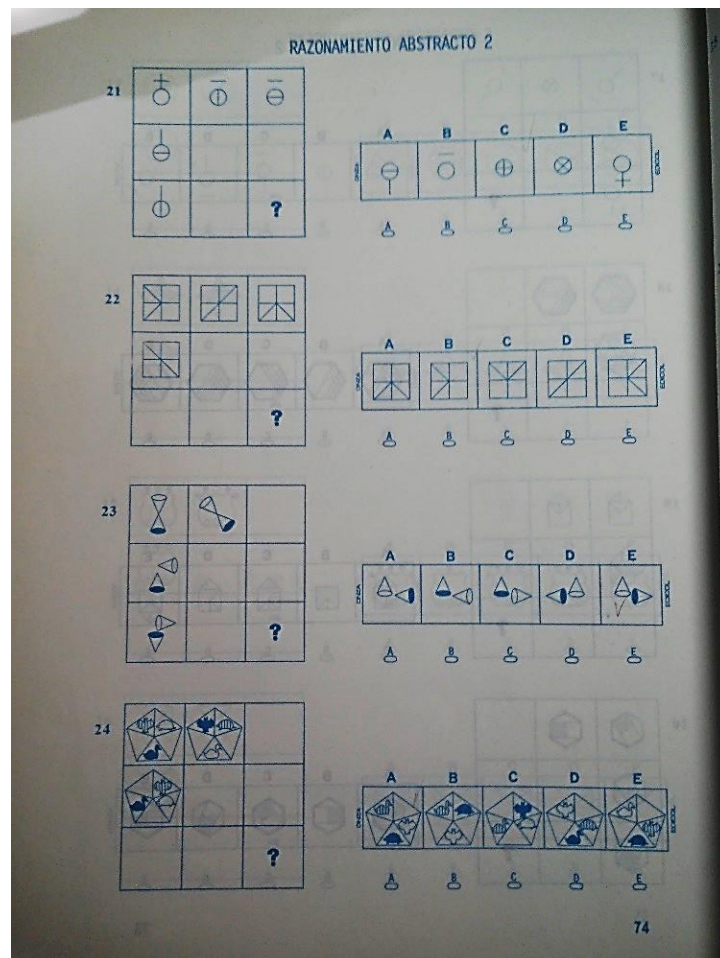


**Figura 24.** Acciones asociadas a las direcciones de la flecha.

### 6.2.5. Lúdica para identificar patrones simples

“El aprendizaje matemático comienza de manera natural muy temprano en la vida con los patrones, que llevan a maneras fabulosas de estimular a su hijo y de reforzar sus destrezas matemáticas tempranas”. (Pellissier, 2016).

Con el propósito de contribuir en la identificación de patrones simples, se ha recurrido al razonamiento abstracto empleado en (Zurira A., 1992), donde se trabaja con pruebas de razonamiento de la forma en la que se muestra en la **Figura 25**, donde se usa la misma grilla que se ha estado trabajando como ventaja, pero dado el público al cual está dirigido este trabajo, se le aplican las correspondientes modificaciones. La idea principal es dibujar un patrón en la cámara en forma de realidad aumentada, donde falte una ficha que el jugador coloca para completar la secuencia (ver **Figura 26**).



**Figura 25.** Fotografía extraída de (Zurira A., 1992).

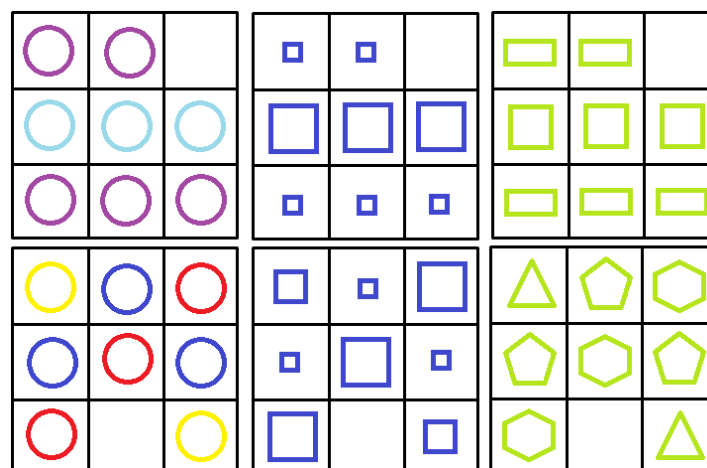
Claramente es necesario ajustar la dificultad del patrón a resolver, para lo cual se han emplean las variables que se han estado trabajando (color, tamaño y forma) y en lugar de tener cuatro ausencias, se reducen a una sola. Por ejemplo,

en la **Figura 26** se muestra un patrón de color en la dirección “x” o derecha con una sola ausencia, en este caso se tiene que la secuencia horizontal es “amarillo, verde y rojo”, por lo que en segunda fila se debe colocar una ficha del primer grupo por su lado de color rojo.



**Figura 26.** Esquema de juego para el reconocimiento de patrones con el color como variable.

Los patrones son de varios tipos, por ejemplo, según el tipo de variable son por color, tamaño y figura geométrica y según su configuración (forma en la que se lee o se entiende) son: horizontal, vertical, diagonal y por simetría y pueden tener dos o tres variables.

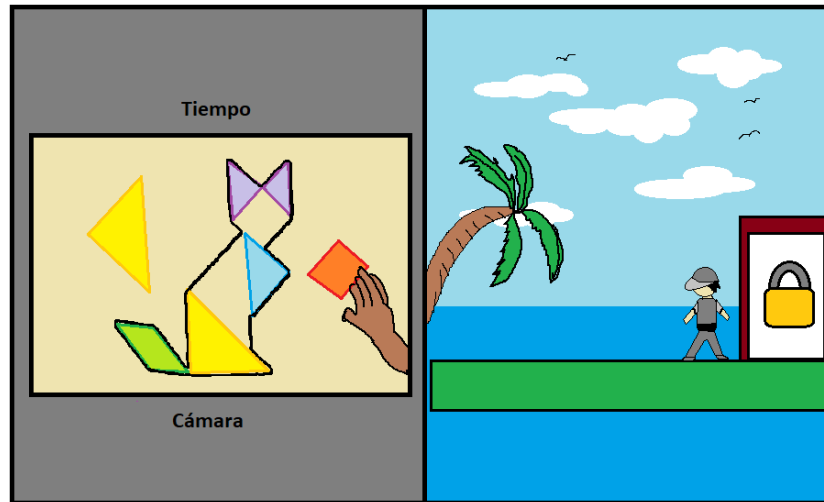


**Figura 27.** Ejemplo de patrones para la aplicación.

La **Figura 27** ilustra varios patrones posibles, la primera fila tiene la misma configuración (patrón vertical de dos variables, “x, y, x”) donde lo único que cambia es el tipo de la variable. La segunda fila muestra tres patrones de tres variables de tipo diagonal “x, y, z, x, y”. Para una mejor comprensión respecto a la elaboración de los patrones remitir al apartado de desarrollo de la aplicación software en arquitectura y procesamiento digital de imágenes.

### 6.2.6. Percepción plana espacial y el tangram

Como componente adicional se ha implementado el juego chino de astucia llamado tangram, con el propósito de ofrecer una alternativa lúdica para niños un poco más grandes (6 años en adelante), para aprovechar los múltiples beneficios con los que cuenta este entretenido juego.



*Figura 28. Esquema de juego con el tangram.*

La actividad consiste en armar la silueta que se dibuja en la cámara dentro de un determinado tiempo (ver **Figura 28**), para ello se emplean las fichas del grupo 4 (ver **Tabla 5**). Cuando la figura se haya completado se desbloquea una puerta y debe salir un personaje que le da una recompensa al jugador y posteriormente se dibuja otra silueta en la cámara para continuar con el juego.

## 6.3. DESARROLLO DE LA APLICACIÓN SOFTWARE

✓ Desarrollar un software capaz de reconocer y analizar figuras geométricas.

Esta es la tercera y última etapa del procedimiento experimental, la cual consiste en última instancia en mostrar las herramientas y lógica en un contexto de software para llevar las lúdicas anteriormente planteadas al entorno de la programación.

### 6.3.1. Entorno de desarrollo

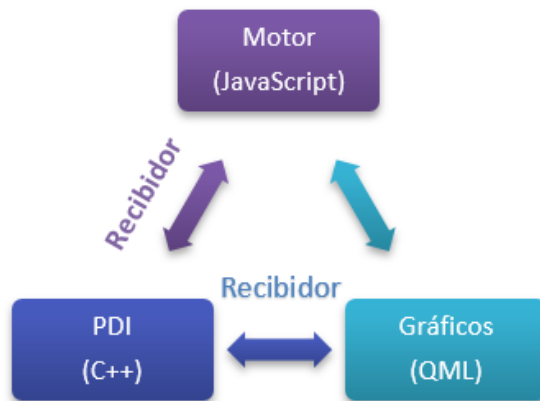
La aplicación está programada en el entorno de desarrollo multiplataforma Qt Creator Community de código abierto y bajo la licencia (L)GPL v3 (visitar: <http://doc.qt.io/qt-5/opensourcelicense.html>), que permite el desarrollo de aplicaciones libres en diversos lenguajes de programación, en este trabajo en

específico se cuenta con tres lenguajes de programación diferentes, lo que convierte a la aplicación en un programa híbrido.

Teniendo en cuenta que la aplicación debía contener un componente de visión por computador y otro gráfico al estilo de un videojuego, entonces era necesario escoger un entorno de desarrollo que permitiera el desarrollo de ambos componentes de manera relativamente fácil y eficiente. Por suerte Qt Creator Community ofrece programación híbrida, lo que posibilita la programación gráfica y el procesamiento digital de imágenes en diferentes lenguajes para disminuir complejidad. En este caso el procesamiento de imágenes se programa en C++ bajo las librerías de OpenCV y el componente gráfico se desarrolla en QML que es una especificación de interfaz de usuario y lenguaje de programación que permite a los desarrolladores y diseñadores crear aplicaciones de gran rendimiento, animadas de forma fluida y visualmente atractivas. QML ofrece una sintaxis muy legible, declarativa, tipo JSON con soporte para expresiones imperativas de JavaScript combinadas con enlaces de propiedades dinámicas (visitar: <http://doc.qt.io/qt-5/qmlapplications.html>). Para la programación gráfica, se ha hecho uso de las librerías diseñadas en QML orientadas a videojuegos y aplicaciones diseñadas por "V-Play" (visitar: <https://v-play.net/>), pero dado que el lenguaje QML necesita de un lenguaje auxiliar que manipule los gráficos creados, es necesario implementar el lenguaje más compatible con QML que es JavaScript, el cual viene a convertirse en el pilar de la aplicación gráfica.

### 6.3.2. Arquitectura de la aplicación

Como se observa en la **Figura 29**, la aplicación consta en resumen de tres componentes principales en un lenguaje diferente cada uno, que interactúan entre sí conjuntamente para cumplir con los objetivos de este trabajo. El componente principal es el motor de la aplicación, este es el que manipula los objetos y animaciones creadas en QML en respuesta a los estímulos obtenidos mediante el procesamiento digital de imágenes (PDI), pero hay componentes gráficos que pueden comunicarse directamente con el PDI sin necesidad de acudir al motor.



**Figura 29.** Esquema general de la aplicación.

Dado que el lenguaje QML es compatible con expresiones de JavaScript, es posible crear partes del motor dentro de los componentes gráficos para los usos más simples, en los casos de más complejidad se debe crear un archivo de extensión “.js” para crear el motor de la aplicación. Esto también facilita la manipulación de los gráficos desde el motor sin necesidad de una clase intermediaria a diferencia la parte de visión por computador, esta debe ser presentada tanto al motor como a los gráficos mediante una clase especial, llamada en este trabajo “Recibidor”, que se encarga de llevar y traer datos de tipo compatible (enteros y cadenas principalmente) entre los componentes mediante funciones declaradas e implementadas dentro de una clase en C++, esto se logra mediante un bucle o “timer” repetitivo bajo un determinado tiempo o una frecuencia de muestreo.

### 6.3.2.1. Muestreo

El muestreo debe ser de tal manera que el PDI tenga el tiempo suficiente para dar una respuesta, es decir que debe ser superior al periodo de procesamiento de las capturas de la cámara, pero a la vez debe ser el mínimo mayor posible porque un muestreo lento generará un “LAG” o retraso muy notable en la experiencia de juego lo que afectará la jugabilidad. Este muestreo puede disminuirse dependiendo de las características de la máquina en que se use la aplicación, por ejemplo, la **Tabla 6** muestra las características y el muestreo usado para la maquina empleada en este trabajo.

**Tabla 6**

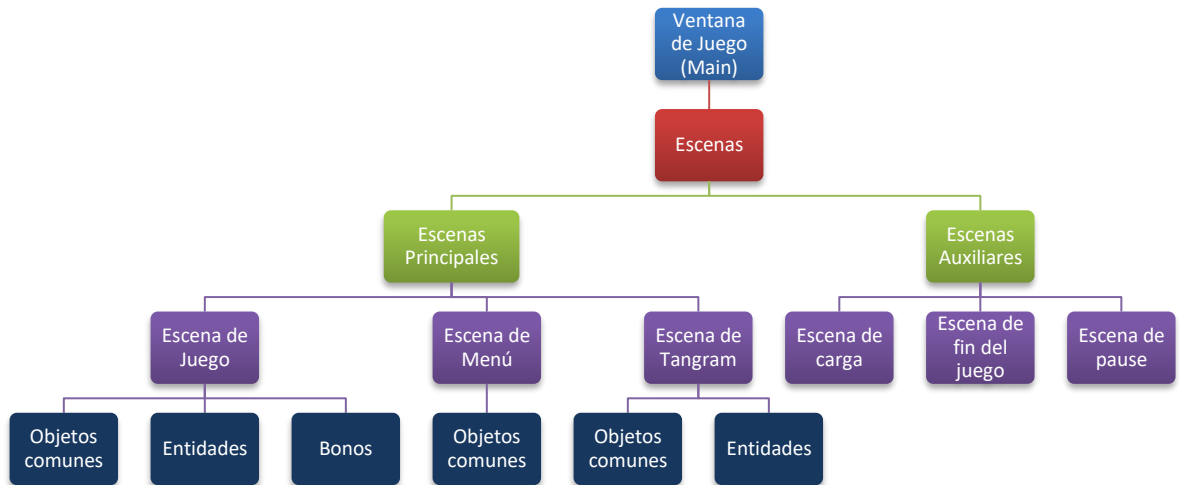
*Características generales de la máquina y del software.*

Características	Valores
Máquina	ASUS X5DIN
Disco duro	300 GB
RAM	4 GB
Procesador	Pentium® Dual Core 2.10GHz
Sistema Operativo	Ubuntu 16.04 64 bits
Qt (Community)	4.7.0 basado en Qt 5.11.1 (MSVC 2015, 32 bit)
OpenCV	OpenCV 3.2.0
Periodo de Muestreo	100 ms

En base a estos valores se elige el periodo de muestreo.

### 6.3.2.2. Componente gráfico

Esta parte está enfocada en los gráficos y animaciones de la aplicación desarrolladas bajo el lenguaje QML.



**Figura 30.** Esquema general de la jerarquía para el desarrollo e implementación de gráficos mediante el lenguaje QML.

Como se observa en el esquema de la **Figura 30**, la aplicación parte de una plantilla principal llamada “ventana de juego”, en esta se emplea una estructura llamada “escena” que define lo que aparece en pantalla. La ventana de juego tiene la propiedad de activar o desactivar las escenas según se requieran y de establecer el fondo musical según sea la escena, esto se logra bajo expresiones de JavaScript en una serie de rutinas que conforman el motor de la aplicación.



Cada escena se crea como un lienzo pre dibujado donde solo se establecen como estáticos a los objetos más básicos como el puntaje, las vidas o la imagen de fondo; los objetos interactivos son previamente definidos, pero llamados durante la marcha como objetos dinámicos que son traídos y borrados de la memoria virtual a medida que avanza el juego.

La escena de juego viene a ser la escena principal en la cual se desarrollan los juegos lúdicos y se llaman los diferentes objetos comunes e identidades. Las entidades son objetos a los cuales se les aplican leyes físicas básicas como la gravedad, las colisiones, fuerzas de fricción, impulso, etc. Los objetos comunes son solo “ítems” como flechas, cajas o cuadros de dialogo entre otros que son usados para desarrollar las escenas. En la escena de juego se llaman muchos objetos y se definen muchas propiedades y funciones que durante la marcha son activadas por el motor de la aplicación en respuesta a las acciones del usuario que ofrecen respuesta en el procesamiento digital de imágenes.

La escena de tangram es independiente de la escena de juego y esta creada como un componente adicional puesto que no forma parte de los objetivos de este trabajo. La escena de menú es para crear las opciones de juego que escoge el usuario además de contener los créditos y opciones como sucede en cualquier otro juego genérico. Finalmente, las escenas auxiliares son simplemente objetos transitorios necesarios para un juego básico.

### **6.3.2.3. Funcionalidad de la aplicación**

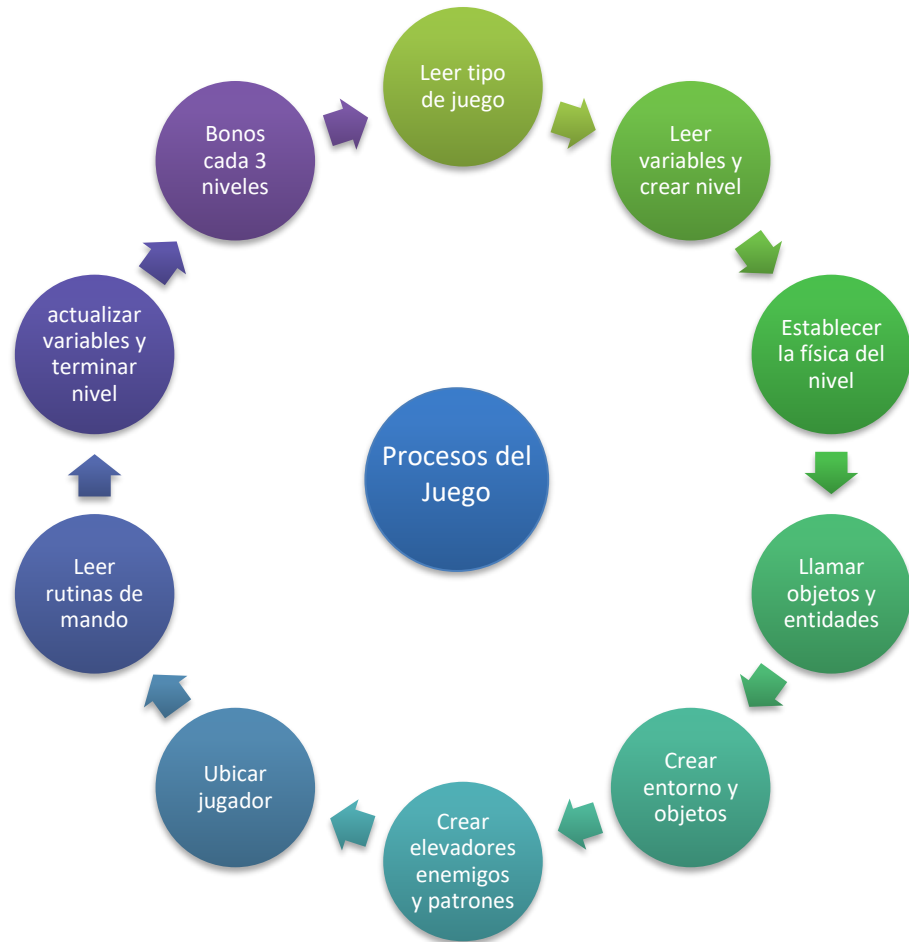
En esta sección se hace énfasis en la estructura lógica y funcional de la aplicación desarrollada bajo funciones en sentencias de JavaScript. El motor se ejecuta en esencia para leer y actuar en base a las respuestas generadas por el PDI y para crear toda la dinámica del entorno gráfico que percibe el usuario o jugador, la primera parte se resume en el diagrama cíclico de la **Figura 31**, en donde se aprecia el orden del proceso para dar respuesta a las acciones que generan estímulos en el PDI, esto se logra mediante el uso de la clase recibidora mencionada anteriormente que entrega los datos necesarios del reconocimiento de las figuras en un vector que almacena: posición relativa en la grilla de la cámara, color de la ficha, tamaño y figura geométrica.



**Figura 31.** Esquema general para la funcionalidad del juego referente al mando de la aplicación.

La estructura para el juego en sí, se presenta en la **Figura 32**, donde se ilustra brevemente el proceso que crea el entorno donde se desarrollan todas las actividades, este entorno depende del tipo de juego que escoja el usuario, puede ser por colores, tamaños o figuras, de esto depende el estilo del paisaje, la propiedad activa de los elevadores y enemigos y las respuestas que deben generarse en base a la lectura del PDI. La idea general es crear varios pisos (ver **Figura 14**) y sus ornamentos automáticamente por cada nivel. Para superar cada nivel se debe llevar el personaje hasta la cima, lo que dará por terminado el nivel y dará la orden para generar el siguiente nivel que será un piso más alto que el anterior. Cada tres niveles se genera un bono, el cual es el juego de las mariposas (ver **Figura 15**).

El algoritmo principal debe leer y actualizar las variables en cada nivel, configurar las leyes físicas simuladas por el motor como la gravedad y las colisiones en un determinado tiempo de muestreo para la lectura de los choques (por defecto es 100ms), luego debe llamar los objetos y entidades necesarios para colocarlos en escena como los objetos de ambientación y los de interacción como los elevadores y los enemigos, posteriormente se ubica el jugador en pantalla y según las respuestas del algoritmo de mando se le atribuyen características que permiten el avance del juego si las acciones del usuario son correctas en el procesamiento; cuando el jugador termina cada nivel se actualizan todas las variables en juego para continuar con la actividad dependiendo del tipo de juego.

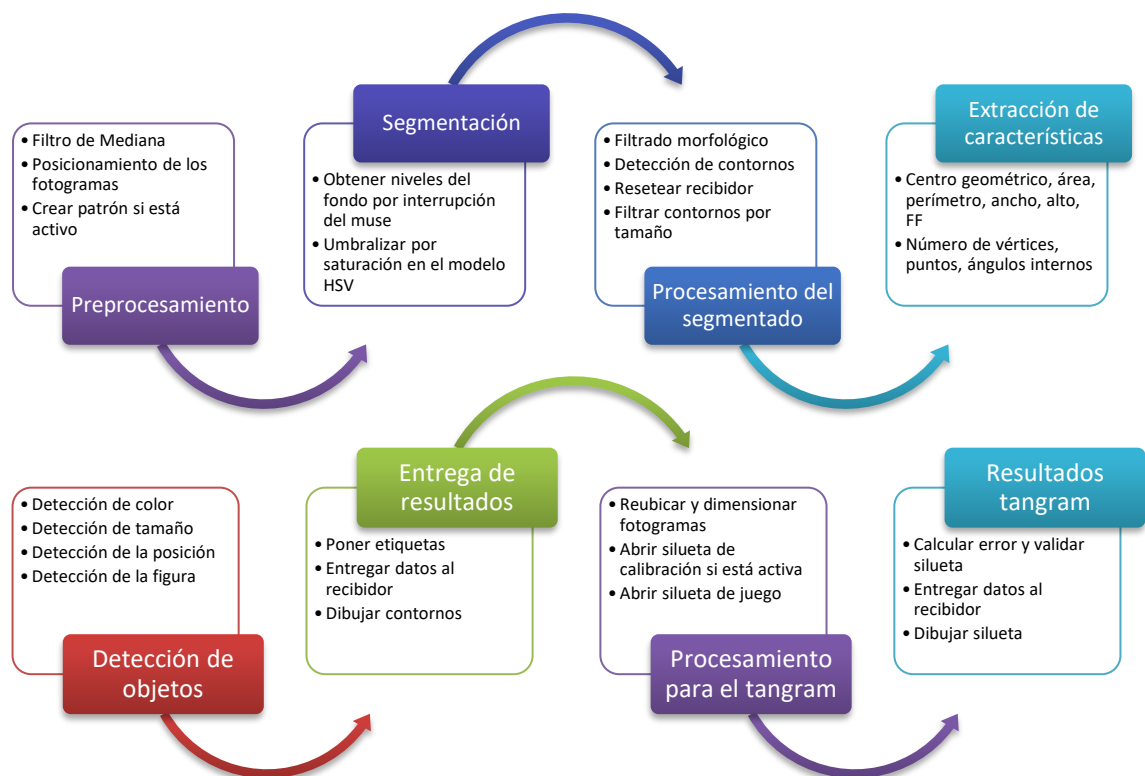


**Figura 32.** Proceso general del juego o algoritmo principal de la aplicación.

El juego de las mariposas o bono de juego, es un nivel simple para dar puntuación y refrescar la experiencia de juego que se genera cada tres niveles para ofrecer puntuación fácil de obtener y probar otro tipo de habilidades motoras.

#### 6.3.2.4. Procesamiento digital de imágenes

En este apartado se procede a presentar de manera general, los procesos más relevantes para el procesamiento digital de los fotogramas entregados por la webcam. En este punto se omite la etapa de captura que ya ha sido establecida previamente en el apartado de “ENTORNO FÍSICO” (ver **Tabla 1**); Las demás etapas del procesamiento se encuentran resumidas en la **Figura 33**, y a continuación se procede a expandir brevemente cada etapa para mayor claridad.

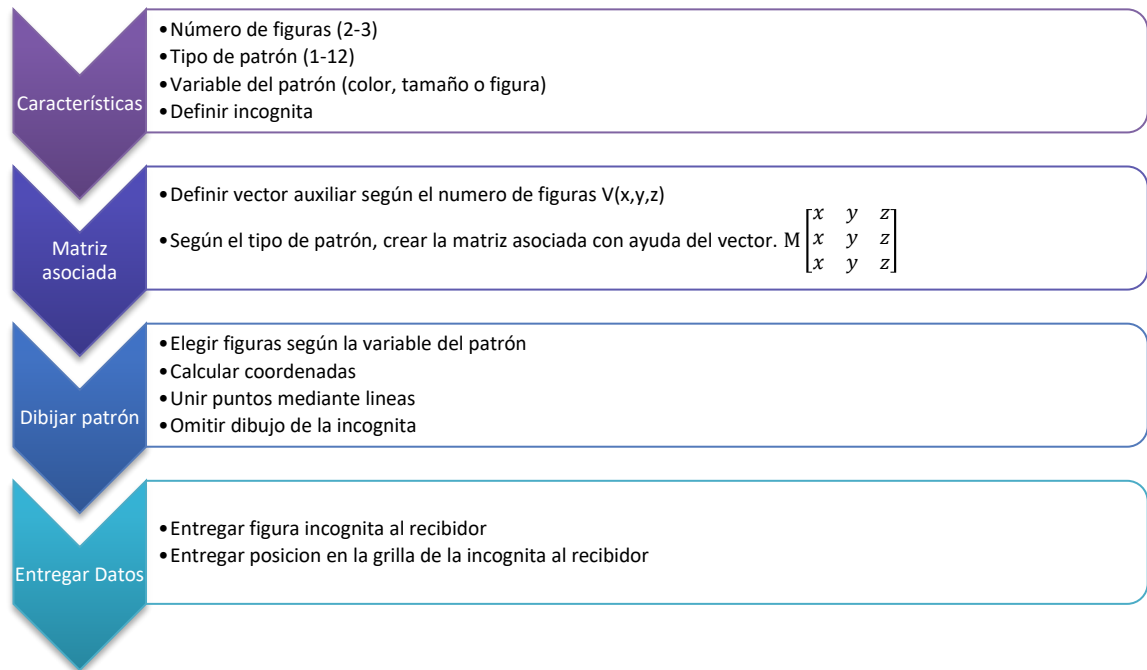


**Figura 33.** Flujo del procesamiento de imágenes para cada fotograma.

### I. Preprocesamiento.

Como etapa previa al procesamiento se balancea tenuemente los niveles de color del fondo que presentan determinado nivel de ruido inherente a la cámara, esto se logra aplicando un filtro de mediana bajo un “kernel” o ventana de 5x5 que ayuda a aplanar el color para disminuir el ruido en la etapa de segmentación. Adicionalmente se posicionan y redimensionan los frames a la mitad de su tamaño original (640x480->320x240), lo que genera un ahorro muy considerable en el procesamiento, específicamente cuatro veces menos operaciones por fotograma, lo que representa un gran aumento en el rendimiento del programa.

La etapa concluye con la solicitud para crear y dibujar determinado patrón en la cámara. La creación de patrones no hace parte del procesamiento como tal, sino que es realmente un componente de realidad aumentada para complementar la lúdica referente al reconocimiento de patrones, este proceso se esclarece en la **Figura 34.**



**Figura 34.** Proceso para la creación y dibujo de patrones.

Los patrones creados bajo este algoritmo o proceso son de la forma que se ilustra en la **Figura 27**, primero se establecen las características para su creación: el número de figuras que pueden ser dos o tres, el tipo de patrón que se refiere a la simetría o al patrón en sí, que es la manera en que se lee el arreglo de las figuras, por ejemplo, la matriz M en la **Figura 34** muestra un arreglo que se lee horizontalmente como “X,Y,Z”, este sería un patrón horizontal. En segunda instancia se crea una matriz asociada al patrón, elaborada a partir de un vector que contiene las figuras posibles, este arreglo matricial le indica a un algoritmo de dibujo, cómo debe representar el patrón en pantalla con la información relacionada al tipo y variable. A continuación, se muestran los tipos de patrones creados para el programa y un ejemplo para su matriz asociada cuando se tiene tres figuras para representar el patrón.

En la **Tabla 7**, específicamente en la matriz asociada al patrón, cada variable representa una figura que se dibuja en la grilla como en la **Figura 27**, pero esta figura puede variar sus atributos según sea la variable que está en juego, por ejemplo si la variable es el color, se tendrán solo círculos que varían de color, en ese caso “X,Y,Z” representan los colores que puede tomar el círculo, en otro caso representan los tamaños de un cuadrado y en tercer caso, representan tres figuras geométricas. Estos atributos se determinan de manera aleatoria para que los patrones estén en constante cambio. Finalmente, las funciones de dibujo toman esta información y representan el arreglo en los fotogramas teniendo en cuenta el vacío para la incógnita, cuya información es transmitida al receptor, para que el algoritmo del videojuego pueda validar si se ha completado el patrón.

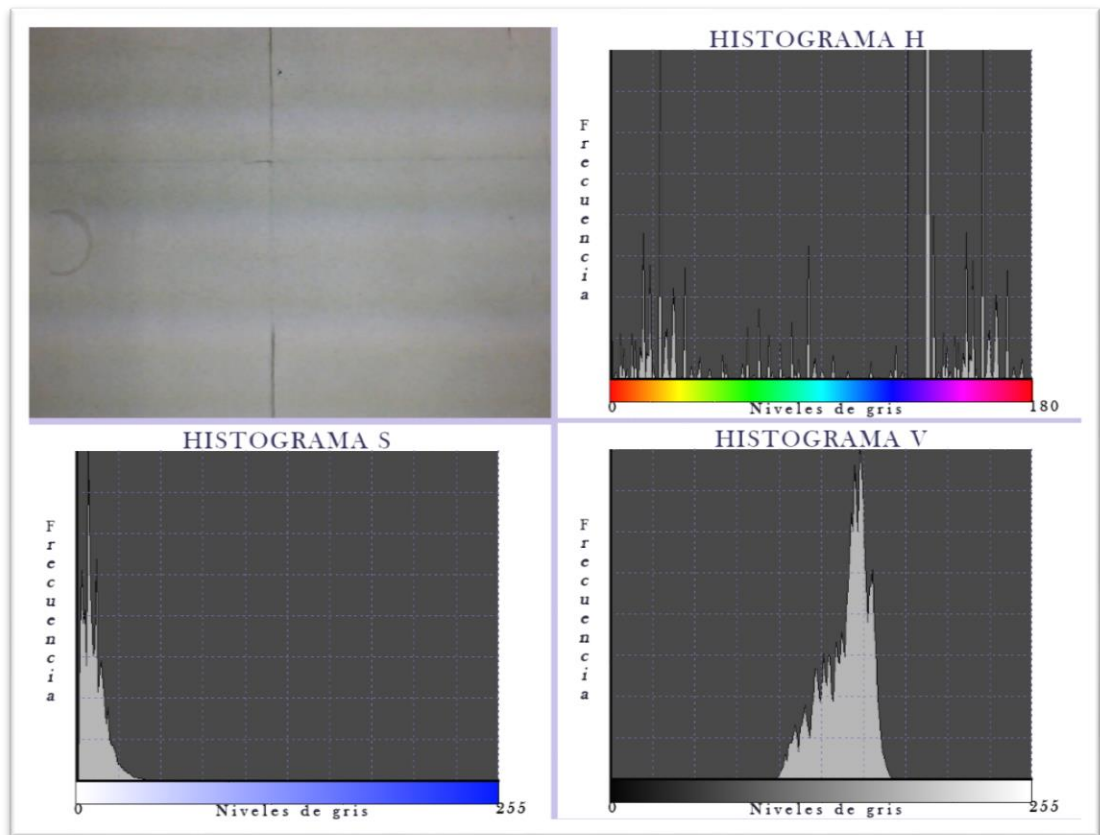
**Tabla 7***Tipos de arreglos y sus matrices asociadas.*

Nombre del arreglo	Matriz asociada (M)
Horizontal	$\begin{bmatrix} x & y & z \\ x & y & z \\ x & y & z \end{bmatrix}$
Vertical	$\begin{bmatrix} x & x & x \\ y & y & y \\ z & z & z \end{bmatrix}$
Diagonal	$\begin{bmatrix} x & x & z \\ x & z & y \\ z & y & y \end{bmatrix}$
Diagonal rotada	$\begin{bmatrix} z & x & x \\ y & z & x \\ y & y & z \end{bmatrix}$
Cruz	$\begin{bmatrix} y & x & y \\ x & x & x \\ z & x & z \end{bmatrix}$
Cruz rotada	$\begin{bmatrix} z & x & y \\ x & x & x \\ z & x & y \end{bmatrix}$
Diagonal 2	$\begin{bmatrix} x & y & z \\ y & z & y \\ z & y & x \end{bmatrix}$
Diagonal 2 rotada	$\begin{bmatrix} z & y & x \\ y & z & y \\ x & y & z \end{bmatrix}$
"X"	$\begin{bmatrix} x & y & x \\ z & x & z \\ x & y & x \end{bmatrix}$
"X" rotada	$\begin{bmatrix} x & z & x \\ y & x & y \\ x & z & x \end{bmatrix}$
"S" especcular	$\begin{bmatrix} x & y & z \\ z & y & x \\ x & y & z \end{bmatrix}$
"S" rotada	$\begin{bmatrix} x & z & x \\ y & y & y \\ z & x & z \end{bmatrix}$

## II. Segmentación.

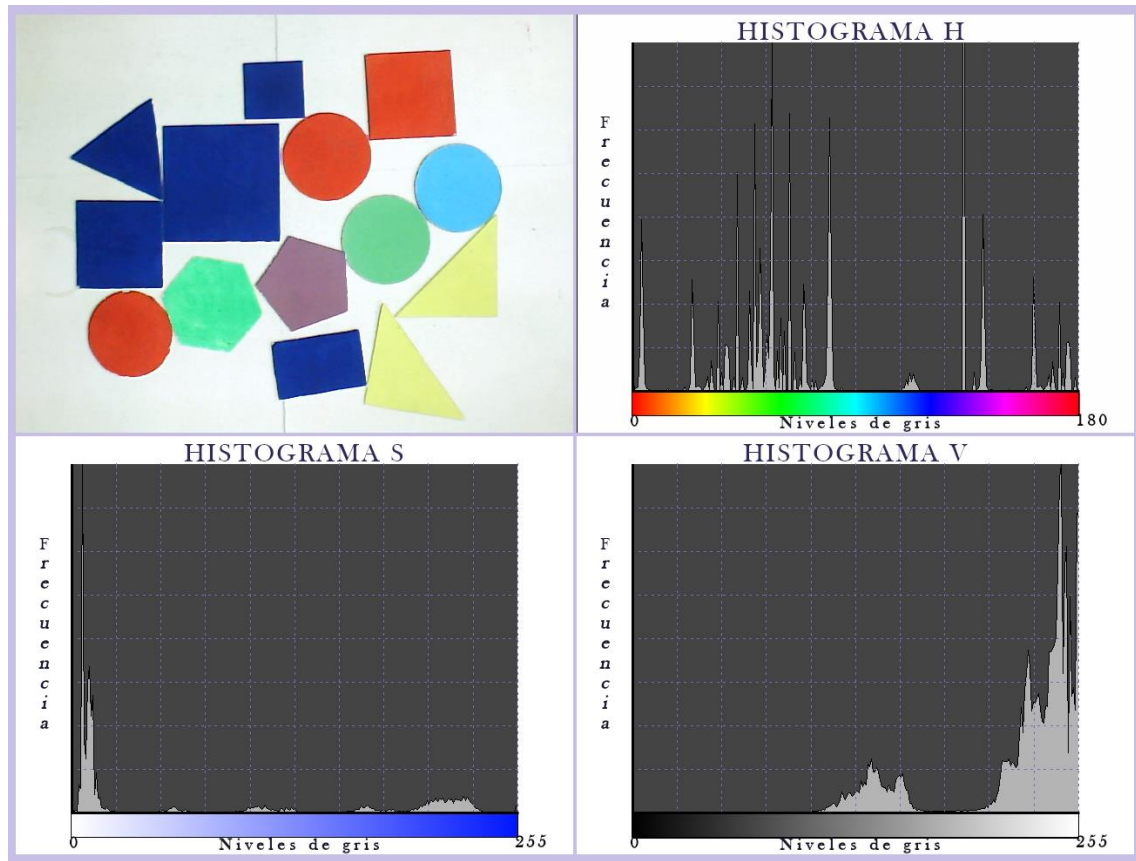
En esta parte del proceso se describen los métodos empleados para binarizar la imagen, en primera instancia es necesario caracterizar el fondo para establecer las herramientas necesarias para la umbralización, para ello es muy útil conocer el histograma en sus diferentes planos de color, en este caso se ha optado por emplear el modelo de color HSV, dado que separa el color y la saturación en

planos diferentes, lo que lo hace más apropiado para segmentar por saturación e identificar colores.



**Figura 35.** Histogramas HSV para el fondo vacío.

La **Figura 35** muestra la respuesta de los histogramas de los planos de color HSV para un fondo que se encuentra limpio; se puede notar que el plano “H” no tiene una tendencia fija, dado que los colores grisáceos pueden tomar cualquier valor en este plano, a diferencia de los planos “S” y “V” que muestran una tendencia a una baja saturación y un brillo medio respectivamente. Ahora se debe analizar la respuesta de los histogramas, cuando se tienen las fichas de juego presentes (ver **Figura 36**)

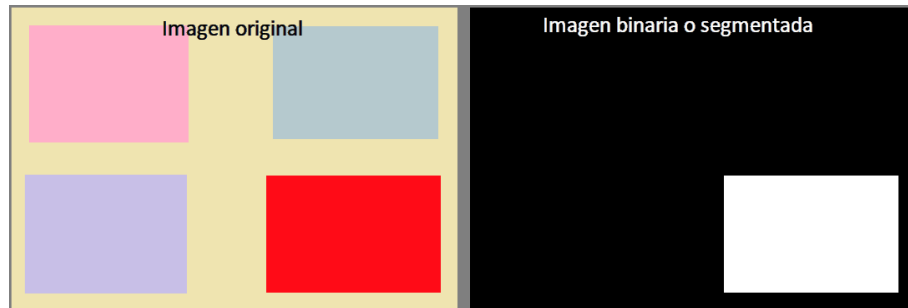


**Figura 36.** Histogramas HSV para el fondo con las fichas.

Se aprecia que la respuesta para el plano “H” sigue siendo complicada de analizar y el Brillo es generalmente medio-alto, pero el plano “S” muestra diferentes saturaciones correspondientes a los diferentes colores y una respuesta mayoritaria en los niveles bajos correspondiente a los pixeles del fondo, aunque en primera instancia no es fácil deducir esto, para ello se cuenta con los histogramas respectivos para los diferentes colores, que se muestran en el **Anexo A**, con esta información se puede concluir que es más conveniente umbralizar en el plano de saturación, ya que el brillo depende de que se cuente con iluminación constante, además de que las sombras podrían generar errores en la segmentación, por ello se opta por emplear los valores de saturación, aunque no unos valores fijos, pues se aprecia en el **Anexo A**, que los histogramas se corren para los diferentes colores y posiblemente también con la iluminación, por ello se deben obtener los niveles de gris del fondo mediante interrupciones del mouse o clicks en diferentes puntos del fondo mediante un promedio de saturación obtenido a través de un kernel de 9x9 para cada click. En total son cuatro clicks que almacenan promedios de saturación, que posteriormente se vuelven a promediar entre los cuatro para obtener un solo valor que representa el nivel de gris promedio correspondiente al plano de saturación del fondo. Con la anterior información se logra umbralizar empleando la función “inRange” de OpenCV que consiste en segmentar a partir de un rango



de valores; si el nivel de gris de un pixel está dentro del rango se asignará un alto (255 o el color blanco) en la imagen binaria y si no será un bajo o cero, lo que se ve como el color negro en la segmentación como se ve en la **Figura 37**.

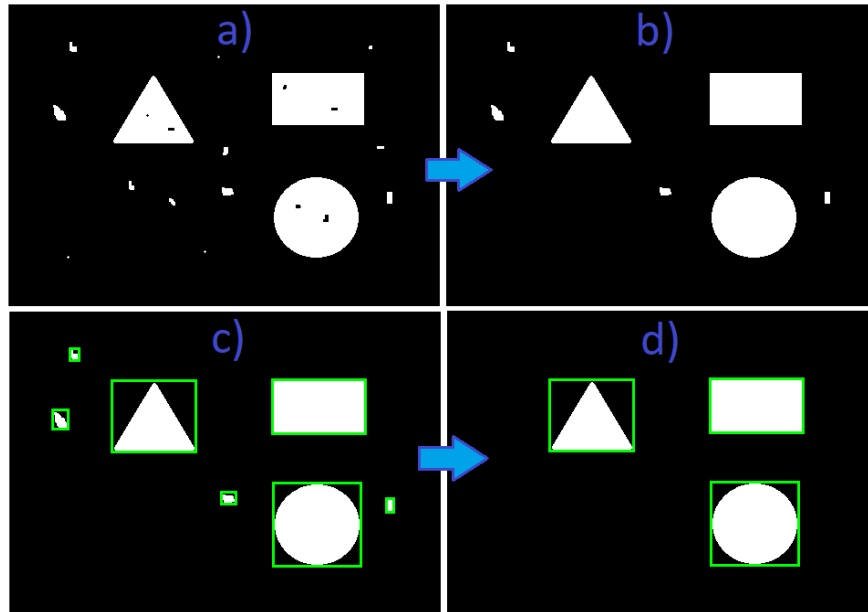


**Figura 37.** Ejemplo de segmentación por saturación; solo el rectángulo más saturado supera el umbral.

### III. Procesamiento del segmentado.

Cuando ya se ha obtenido la imagen binaria, se procede primeramente a filtrar los defectos resultantes de la umbralización, esto se logra aplicando las operaciones morfológicas apertura y cierre con un kernel de 3x3, la primera elimina el ruido blanco de la zona negra, y la segunda suprime las manchas negras de la zona blanca; este filtrado solo elimina las manchas o ruido que son más pequeñas que el kernel; una ventana más grande genera deformaciones en la imagen.

Posteriormente al filtrado morfológico, se obtienen los contornos de la imagen empleando la función "findContours" de OpenCV que almacena cada contorno o región continua en un arreglo vectorial que se filtra luego por tamaño para eliminar esas regiones pequeñas que sobreviven al filtrado anterior. Debido a que las siguientes etapas son para obtener los datos que se envían al receptor, es necesario que éste se encuentre vacío para que los datos no se sobrescriban y tengan su último valor, ya que un valor vacío también es información; implica que no hay una ficha presente en determinada región. Esta etapa se resume en la **Figura 38**.



**Figura 38.** Procesamiento del segmentado; a) segmentación sin procesar, b) filtrado morfológico, c) detección de contornos, d) filtrado por tamaño de contorno.

#### IV. Extracción de características.

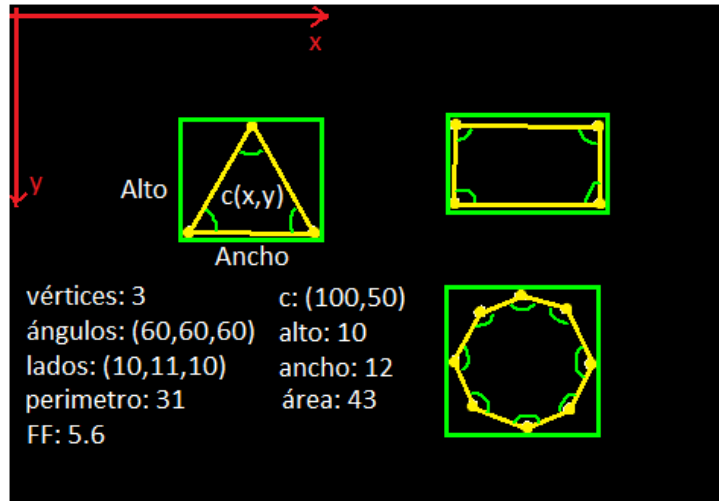
Como se ve en la **Figura 33**, se toman dos grupos de características, el primero se extrae directamente de cada contorno a través del conteo de píxeles internos y periféricos mediante funciones simples de OpenCV, además de que se le asocia a cada región un rectángulo mínimo que la contiene. Estas herramientas permiten calcular: área, perímetro, ancho, alto, centro geométrico y un factor de forma:

$$FF = \frac{40\pi A}{P^2} \quad (1)$$

Con A: área, P: perímetro.

En la medida en la que el factor de forma se aproxime a diez, el contorno se aproxima a un círculo, esto sirve como criterio para la detección de la figura más adelante.

El segundo grupo de características se obtiene a partir de la aproximación de la región a un polígono mediante la función “approxPolyDP” como en la **Figura 5**. La función permite conocer la cantidad de vértices y sus coordenadas, lo que permite el cálculo de las distancias y ángulos internos del polígono aproximado mediante operaciones básicas de álgebra lineal (ver **Figura 39**).



**Figura 39.** Ejemplo ilustrativo (no real) de la extracción de características para un triángulo en un segmentado ya procesado.

## V. Detección de objetos.

Cuando ya se han extraído las características para cada contorno se procede con la detección. En primer lugar, se detecta el color mediante una matriz de 9x9 ubicada a partir del centro geométrico del contorno, cuyos píxeles se procesan con la distribución estadística de mediana para obtener el nivel de color del plano "H". El tamaño se obtiene directamente del área, aunque simplificando a tres medidas solamente: grande, mediano y pequeño (máximo 2900, 1300, 600 píxeles respectivamente). La posición se toma a partir del centro geométrico, aunque ésta se simplifica a solo nueve estados según la grilla del juego (ver **Figura 40**).

0	1	2
3	4	5
6	7	8

**Figura 40.** Posiciones de una figura en la grilla del juego.

La detección relativamente más compleja es la respectiva a la figura geométrica, para lograrlo se ha empleado el método del árbol de decisión como se ve en la **Figura 41**, donde se aprovecha el número de vértices como descriptor fundamental y el hecho de que se deben detectar solo un determinado grupo de figuras ya definidas anteriormente (ver apartado de las características para las figuras)



**Figura 41.** Árbol de decisión empleado para detectar la figura.

Hasta este punto ya se ha detectado cada ficha con su color, tamaño posición y figura geométrica, aunque debe resaltarse que la detección no es perfecta y puede tener ligeras fallas a modo de ruido, aunque muy sutiles.



**Figura 42.** Ejemplo ilustrativo de la detección de objetos para un triángulo.

## VI. Entrega de resultados

Debido a que el apartado gráfico se comunica con el PDI a través de un receptor, es necesario que éste se actualice con los datos obtenidos en la detección, para ello se define en el receptor un arreglo vectorial que en cada posición almacena tres datos correspondientes a la figura, el color y el tamaño de la siguiente forma:

$$\text{recibidor.datos}[\text{posición}] = (\text{figura}, \text{color}, \text{tamaño}) \quad (2)$$

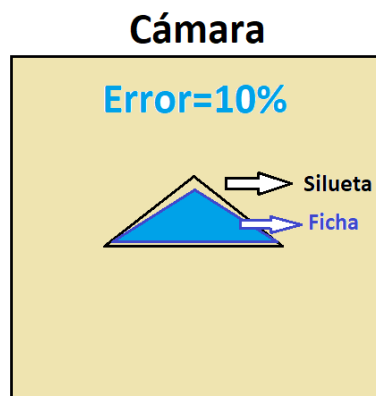
En términos de programación, el receptor viene a ser el objeto de la clase mediadora para la intercomunicación entre módulos (gráfico y PDI), “datos” es el arreglo miembro del objeto receptor, que almacena en cada posición (de 0-9, ver **Figura 40**) las características de la ficha que allí se encuentra, por ejemplo:

$$\text{datos}[4] = (\text{triangulo}, \text{rojo}, \text{pequeño}) \quad (3)$$

La anterior ecuación indica que en la grilla número cuatro se encuentra un triángulo con dichas características. Para finalizar esta etapa, se dibujan y etiquetan los contornos con la información obtenida para verificar que la detección realizada es correcta, de esta manera queda clara la metodología experimental para dar cumplimiento al primer objetivo específico.

## VII. Procesamiento para el tangram.

En esta sección se presentan las acciones necesarias para que la actividad de juego con el tangram sea posible. La primera tarea consiste en reubicar y redimensionar los frames, puesto que la figura detectada en este caso consiste en el contorno asociado a una de las siluetas del conocido juego de astucia. Debido a que la aplicación tiene una opción de calibración para que la figura armada encaje en la silueta, se debe abrir la plantilla de calibración si el modo de calibración está activo; la plantilla es simplemente la silueta asociada a la ficha más grande del juego tangram, y en base a esto se debe configurar la cámara para que el empalme sea el mejor (ver **Figura 43**).



**Figura 43.** Calibración de la ubicación de la cámara para el tangram.

Cuando no es necesario calibrar la disposición de la cámara, se procede a dibujar las siluetas del juego como en la **Figura 28** y se calcula el error entre la plantilla en cámara y la figura armada de la siguiente manera:

$$\%error = \frac{100}{255Area_{Contorno}} \sum |plantilla - contorno| \quad (4)$$

Dependiendo de este coeficiente de error se valida la figura armada en cámara y si es correcta se envía la información al receptor y se procede a dibujar la siguiente silueta. La **Figura 44** muestra algunas de las plantillas usadas para el juego.



**Figura 44.** Plantillas empleadas para dibujar las siluetas del tangram.

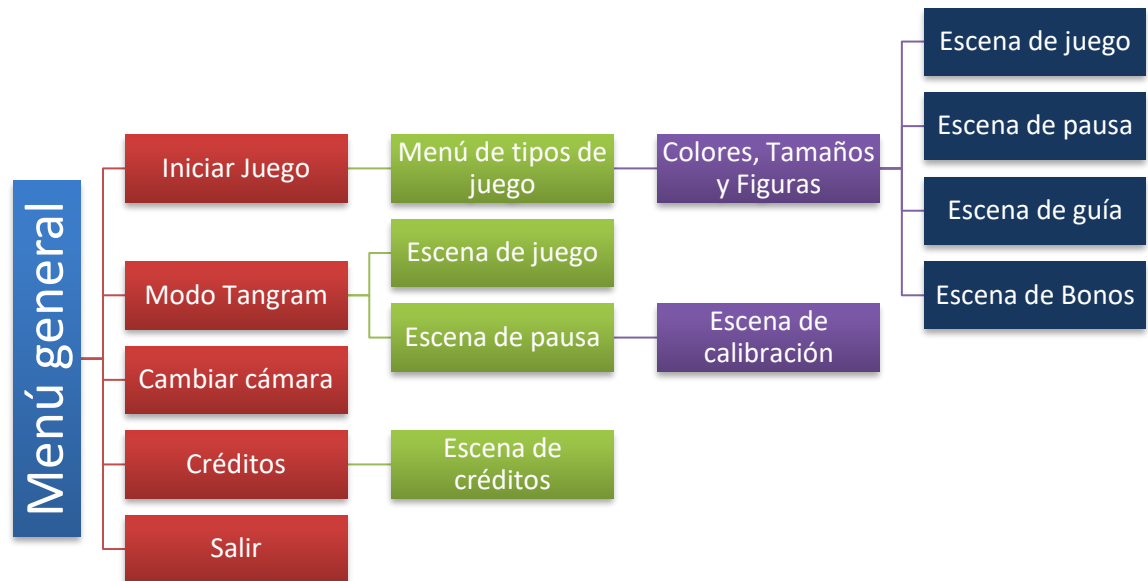
Con esta etapa concluye el procesamiento digital de los fotogramas teniendo en cuenta que se han omitido varios detalles técnicos con el propósito de dar a conocer solamente los procedimientos más generales de todo el proceso.

### 6.3.3. Interfaz gráfica de usuario

✓ Crear una interfaz amigable con el público infantil en el rango establecido

Con la finalidad de crear una interfaz gráfica agradable al usuario y acorde a las lúdicas propuestas, se establece el diseño como el de un videojuego estándar en 2D, cuya raíz es un menú general que permite el acceso a las funciones del programa. Como se ve en la **Figura 45**, el menú ofrece como primera opción iniciar el juego, lo que despliega un submenú, para escoger el tipo de juego y luego de elegir se inicia la escena de juego que está ligada a una escena de

pausa, otra escena para la transición de niveles o carga, en donde se ofrecen algunas guías y la escena para los bonos que se alternan entre los niveles.



**Figura 45.** Estructura básica del menú general.

La escena para el modo Tangram es similar a la del juego ya establecida, solo que se le asigna un mayor espacio a la cámara y cuenta con una configuración o escena de calibración. La opción para cambiar la cámara es simplemente para aquellos casos en que se cuente con una máquina de más de dos webcams y su función es simplemente cambiar el número del dispositivo usado en OpenCV.

Finalmente, la opción de los créditos es usada principalmente para referenciar las fuentes que permitieron obtener los “assets” o elementos necesarios para la creación de la aplicación en general, tales como los sonidos texturas, imágenes, sprites y demás elementos artísticos usados bajo licencia de libre uso.

La estética de la interfaz de usuario está diseñada, en base a las aplicaciones para smartphones, por lo que los niños pequeños las encuentran visualmente muy atractivas por lo que se emplean colores llamativos animaciones, transparencias y sonidos de interacción y en con estos elementos en mente se ha diseñado y desarrollado cada escena en la aplicación.

## 7. RESULTADOS Y ANÁLISIS

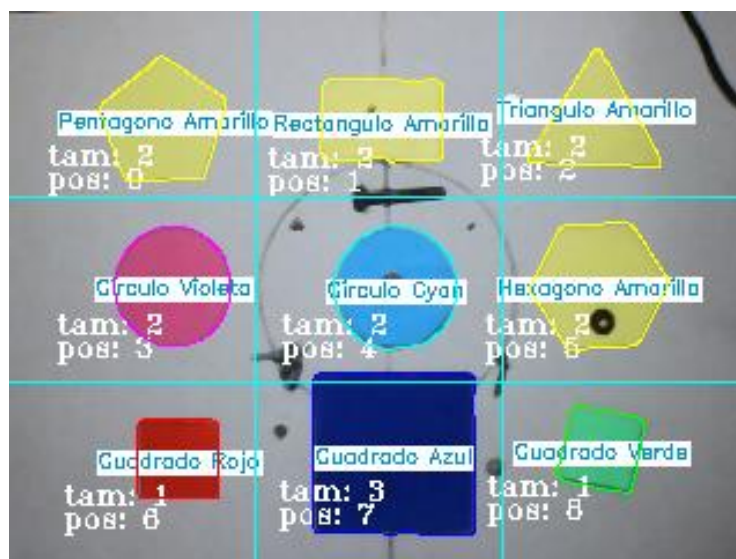
En esta sección se presentan los principales logros en la aplicación respecto a sus tres componentes esenciales: procesamiento digital de imágenes, gráficos e interfaz y lúdicas.

### 7.1. RESULTADOS DEL PROCESAMIENTO DIGITAL DE IMÁGENES

A continuación, se presentan los resultados de aplicar los procesos descritos en el apartado de procedimiento experimental para el desarrollo de la aplicación software, en concreto, refiriéndose al reconocimiento de figuras con sus características.

#### 7.1.1. Reconocimiento de formas, tamaños y colores

Como puede evidenciarse en la **Figura 46**, se logró reconocer satisfactoriamente las formas geométricas propuestas para las lúdicas con sus colores y sus tamaños, aunque esta última información no necesariamente se despliega en cámara. Esto se consigue tanto en la resolución de 640x480 como en 320x240.

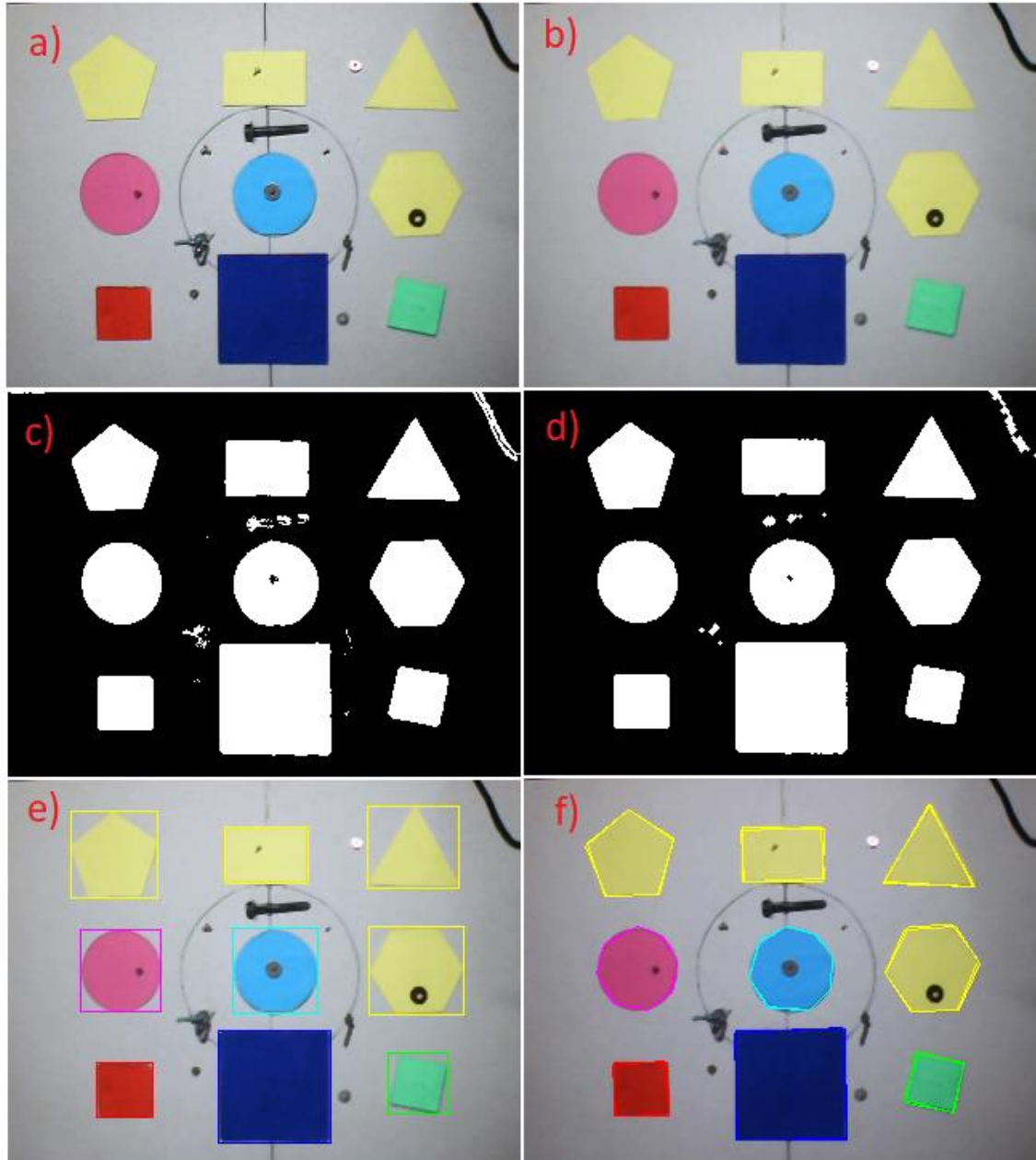


**Figura 46.** Reconocimiento general de figuras.

La anterior captura muestra la figura, el color, el tamaño y la posición en la grilla para un grupo de nueve fichas colocadas entre objetos que representan ruido o información no deseada para dar evidencia del funcionamiento de las etapas del



procesamiento como se muestra en la **Figura 47**, la cual permite apreciar los pasos previos para llegar a los resultados de la **Figura 46**. La estructura del programa que hace el PDI se muestra en el **Anexo B**.



**Figura 47.** Resultados de las etapas previas al reconocimiento; a) imagen sin procesar, b) filtrado de mediana, c) segmentación sin procesar, d) filtrado morfológico, e) detección de contornos, f) aproximación de los contornos a polígonos.

Cabe resaltar que el reconocimiento de las figuras, depende en gran medida de la iluminación, puesto que una iluminación dura puede generar que el algoritmo de Douglas-Peucker aproxime el contorno con una cantidad de puntos errada, por tal razón es necesario que la iluminación sea suave o controlada como se

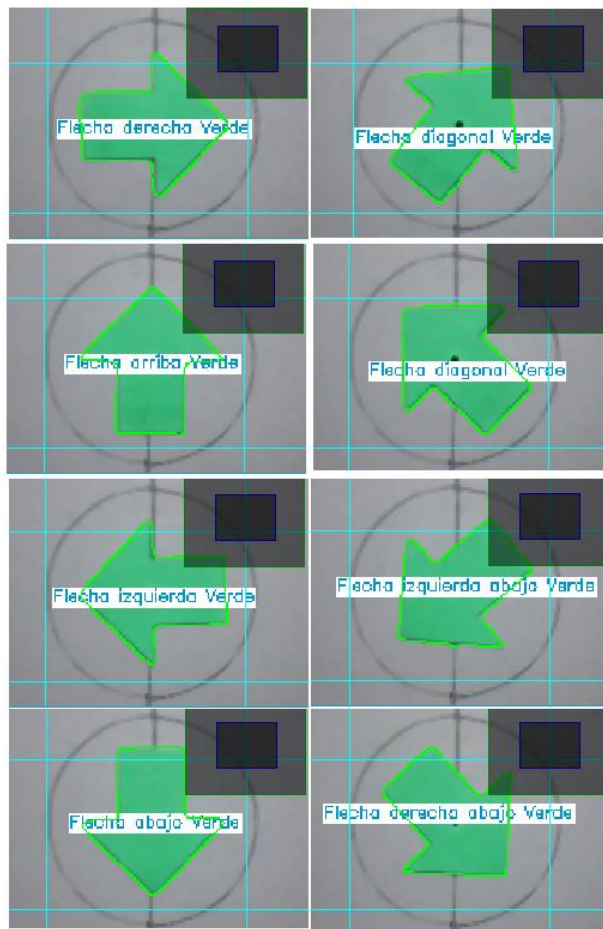
muestra en la **Figura 46** para obtener un resultado como el mostrado. Los principales errores se presentan cuando las figuras son similares ante la cámara por la baja resolución, por ejemplo, el hexágono puede confundirse con un pentágono y al revés, esto sucede a menudo si la iluminación genera muchas sombras, por lo que se deja en claro que el reconocimiento no es perfecto y se puede mejorar para que el programa sea más robusto, pero hasta este punto basta con una iluminación correcta para no tener dificultades.

Una dificultad adicional durante el procesamiento para las figuras se generó porque el algoritmo de Douglas-Peucker aproxima bajo un margen de error o  $\epsilon$ , este parámetro se establece de tal manera que la aproximación sea brusca o fina, es decir, si se quiere aproximar con muchos puntos o pocos, y dadas las necesidades de este trabajo, era necesario emplear una aproximación poco fina porque las figuras tienen ángulos pronunciados y no más de seis, pero esto representa un problema para el círculo que es una figura de borde suave y queda aproximada a un heptágono o hexágono como se ve en la **Figura 5**, pero esto se soluciona empleando el factor de forma ya establecido como se ve en el árbol de decisión de la **Figura 41**, de esta manera se complementa el algoritmo de aproximación de puntos y se obtiene un buen resultado.

### **7.1.2. Reconocimiento de la flecha y su direccionalidad**

Con las figuras reconocidas anteriormente se le otorgan atributos al jugador como su color, tamaño o figura según sea la actividad, sin embargo, es necesario reconocer la flecha y su dirección para otorgar acciones al jugador.

La **Figura 48** muestra que se ha conseguido reconocer la flecha en cada una de sus direcciones para lograr implementar las acciones propuestas en la **Figura 24**, esto se consiguió gracias al algoritmo de Douglas-Peucker, que aproxima el contorno de la flecha a sus siete puntos básicos y tomando el punto de la punta y el punto medio de la cola, se mide el ángulo de la recta que pasa por estos dos puntos respecto a la horizontal y posteriormente se distribuye en rangos separados cada cuarenta y cinco grados para reconocer las direcciones básicas.



**Figura 48.** Reconocimiento de la flecha y su dirección.

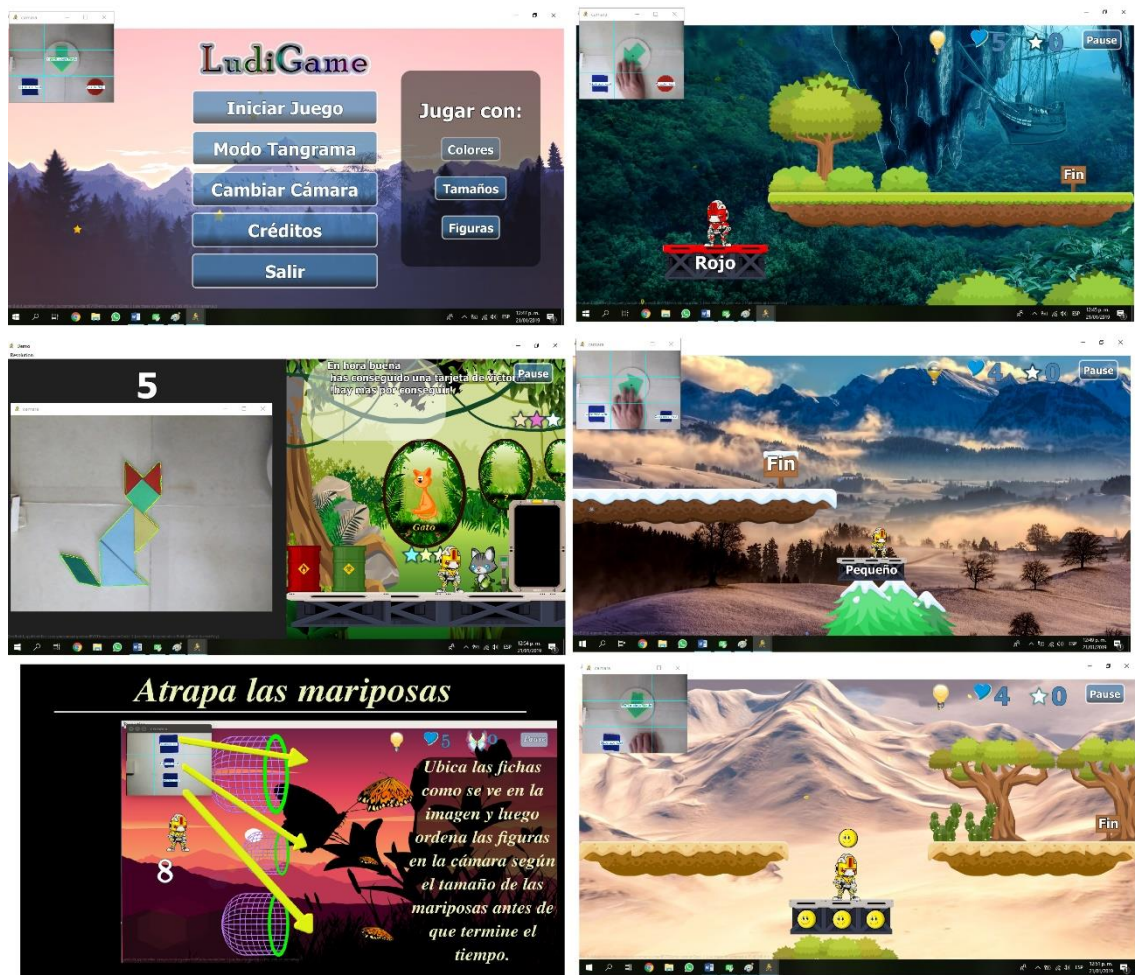
La principal dificultad en el reconocimiento de la flecha se presenta cuando el usuario la gira en un punto donde el ruido del procesado la ubica en un fotograma en una dirección y en el otro en la dirección contigua, es decir que el ángulo está en medio de dos rangos vecinos, la solución a dicho problema es tan simple como mover un poco la flecha dependiendo de la acción que se necesite. Este reconocimiento puede llegar a mejorarse si se prueban otras técnicas como el factor de forma o la comparación con una plantilla, pero ante los requerimientos de este trabajo, los resultados obtenidos son suficientes, aunque mejorables.

## 7.2. RESULTADOS REFERENTES AL COMPONENTE VISUAL DE LA APLICACIÓN

En esta parte se presentan capturas de la aplicación mostrando lo más relevante referente a la programación de gráficos y multimedia mediante el lenguaje QML y JavaScript, principalmente la interfaz de usuario y el desarrollo de las actividades lúdicas propuestas.

### 7.2.1. Interfaz gráfica de usuario

Con la finalidad de crear una interfaz agradable para el público infantil, se emplearon colores llamativos, transparencias y diversas animaciones inspiradas en la apariencia de las aplicaciones para smartphones.



*Figura 49. Algunas escenas de la aplicación.*

La **Figura 49** muestra un resumen de la apariencia de la aplicación, en la que se puede notar que se ha escogido como personaje principal a un robot y como



nombre de la aplicación, “LudiGame” que quiere decir, juego lúdico. Se puede apreciar también que en las escenas de juego aparece una bombilla amarilla que se llena de luz a manera de fluido. El propósito de este indicador es medir indirectamente la fluidez de la aplicación, esto se consigue calculando el periodo de procesamiento por fotograma en el aparatado de PDI mediante “timers”, esto se debe al hecho de que la cámara varía la velocidad del obturador para permitir mayor paso de luz al sensor fotosensible cuando la iluminación es baja en intensidad, esto genera un retraso en el procesamiento de los fotogramas que a su vez genera una disminución en la fluidez de los gráficos o animaciones, es decir que el rendimiento de la aplicación depende de la iluminación en cierta medida, esta es la razón por la cual el indicador es una bombilla.

Una parte importante de la aplicación es la música, la cual debe ser suave y alegre, por fortuna se encuentra en la web diversos bancos de música de libre uso y particularmente en este caso se han empleado algunas canciones publicadas por el sitio web Bensound.com que cuenta con una variedad de temas agradables para pequeños y grandes.

Aprovechando el hecho de que Qt es un entorno de desarrollo multiplataforma, se ha configurado la aplicación para que funcione en Windows y Ubuntu, e incluso hay una versión del juego en desarrollo para Android como se muestra en la **Figura 50**.



**Figura 50.** Desarrollo del juego en varias plataformas.

Los assets para la aplicación se obtuvieron bajo licencia de libre uso, principalmente gracias a los sitios web: [www.gameart2d.com](http://www.gameart2d.com), [www.pixabay.com](http://www.pixabay.com) y [www.vecteezy.com](http://www.vecteezy.com).

### 7.2.2. Lúdicas

Se presentan en esta sección algunas tomas de como resultaron las actividades lúdicas propuestas luego de ser implementadas en el software.

### 7.2.2.1. Lúdicas con colores

La **Figura 51** muestra el desarrollo de las actividades con colores, en donde se aprecia como el jugador se enfrenta a diferentes retos que le exigen manejar y asociar los colores para poder avanzar, desplegando en cámara una silueta de ayuda para saber dónde ubicar la ficha correcta cada vez que se sube al elevador. Cuando la calabaza colisiona con un proyectil del jugador, ésta se quedará inmóvil siempre que la bala y su traje sean del mismo color, en dicho caso se desactivan las colisiones para este personaje que obstaculiza el camino. Las mariposas son un juego de relacionar y ordenar que exige que cada vez se haga en un menor tiempo, esto fomenta la motricidad fina y la coordinación mano ojo entre otras cosas.



**Figura 51.** Actividades con colores.

### 7.2.2.2. Lúdicas con tamaños

Al igual que con las actividades anteriores, los juegos con tamaños, tienen la misma estructura, solo que en elevador aparece una sombra que le indica el tamaño del jugador para que el elevador se active, además de esto se dibuja la silueta en cámara para que el niño la ubique la ficha en el lugar y con el tamaño correcto. Las calabazas aparecen en diferentes tamaños y similarmente al caso anterior, los proyectiles deben ser del tamaño que tiene el enemigo para que éste se detenga y le deje avanzar. El juego de las mariposas es como el anterior, pero con el tamaño como variable (ver **Figura 52**).



*Figura 52. Actividades con tamaños.*



### 7.2.2.3. Lúdicas con figuras

Siguiendo con la estructura anterior, se le asocia al jugador, al enemigo y a los elevadores una figura geométrica para que se complete el reto como en las actividades anteriores, la diferencia se marca en el juego de las mariposas (ver **Figura 53**) donde la asociación y el orden de las figuras se hace horizontalmente para variar un poco, en esta actividad las mariposas traen diferentes fichas y las sueltan en los botes, y si el vote es correcto, las fichas entran, de lo contrario rebotarán.



**Figura 53.** Actividades con Figuras.



#### 7.2.2.4. Lúdicas con patrones simples

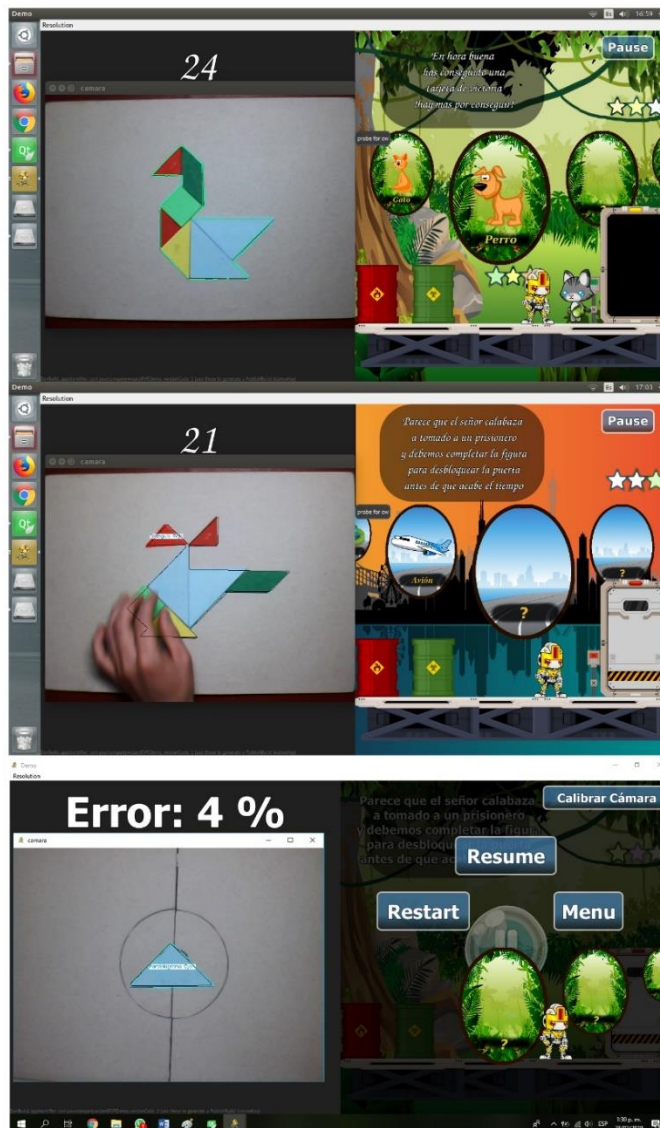
En pro de fomentar el reconocimiento de patrones, estos se presentan a lo largo del juego ocultos en cajas que otorgan estrellas si el patrón se resuelve correctamente. La **Figura 54** muestra algunos patrones en los distintos modos de juego, éstos se despliegan en cámara y desactivan toda acción en el jugador hasta que resuelva el patrón; se empieza por los patrones más simples que en este caso son los horizontales y verticales de dos figuras, si la figura colocada en el espacio faltante es errónea, el algoritmo genera otro patrón de la misma clase al azar, esto se hace con el propósito de que el jugador se pueda equivocar y aprender de ello, esto fomenta la retroalimentación del niño.



**Figura 54.** Actividades con patrones.

### 7.2.2.5. Desarrollo del tangram

Debido a los múltiples beneficios del tangram en los niños, se ha reservado una parte de la aplicación para este juego de astucia como componente adicional o extra. En la **Figura 55** se ve que se ha desarrollado principalmente para dos clases de siluetas; animales y objetos de la ciudad. En las lúdicas anteriores vasta con ubicar la cámara a la altura establecida, pero en este caso, dada la comparación de píxeles para determinar el error, es necesario calibrar la posición y orientación de la cámara, para ello se usa la ficha más grande del tangram como se había mencionado anteriormente. De no calibrarse el juego puede funcionar, pero con mayores posibilidades de fallos.



**Figura 55.** Implementación del tangram.

### 7.3. RESULTADOS DE IMPLEMENTACIÓN

Anteriormente se presentaron resultados referentes al desarrollo de la aplicación; en esta sección se presentan resultados referentes al impacto de la aplicación sobre el público infantil sobre el cual se probó el software. Estas pruebas se realizaron en el Hogar del Instituto Colombiano de Bienestar Familiar los Jilgueros patrocinado por la Asociación de los Hogares Comunitarios de Bienestar, vereda Calibío y a cargo de la madre comunitaria Diana Milena Iles Gómez.

#### 7.3.1. Jugabilidad

En este contexto se hace referencia al término de jugabilidad como la experiencia del niño al interactuar con el juego mediante las fichas, con esto en mente, se realizó una prueba para determinar qué partes o controles del juego se les dificultaban o se le hacía fácil a los pequeños para realizar mejoras según sea la adaptación que tengan los niños con la aplicación.



*Figura 56. Prueba de jugabilidad.*

Como se ve en la **Figura 56**, para esta prueba se empleó una mesa blanca como fondo para las figuras, la iluminación era con luz natural y no hubo problema con el reconocimiento. Respecto a la jugabilidad, los pequeños pudieron manejar el juego, aunque algunos presentaron algunas dificultades motrices para ubicar la figura en la grilla de la cámara y para asignar acciones al personaje mediante la flecha, pero de manera general la mayoría de niños pudieron superar los retos del juego y les fue muy agradable la interfaz de usuario, las lúdicas y la música.



Para mayor validez, se pueden ver algunos videos de los pequeños jugando, disponibles en <http://bit.ly/2GP0yXn>.

### 7.3.2. Mejoras

En base a la prueba de jugabilidad se realizaron varias mejoras técnicas poco relevantes para ser mencionadas, pero además también se hicieron algunos cambios más importantes, se ajustaron tiempos para las mariposas y los elevadores según los niños los necesitaban, también se prolongó el juego de las mariposas que les fue muy agradable y lo más importante fue que se adicionó un tutorial en la aplicación para facilitar el manejo de la direccionalidad de la flecha y la grilla de la cámara.



**Figura 57.** Tutorial del juego.

En la **Figura 57** las primeras capturas muestran cómo se debe ubicar la flecha para que el personaje actúe de diferentes maneras, esto se hace bloqueando todas las acciones del personaje para posteriormente ir las desbloqueando a medida que avanza el tutorial. En las últimas dos capturas se ve una parte del

tutorial referente al manejo de la grilla, para ello se coloca una silueta que va saltando por la grilla cada cierto tiempo que a su vez va disminuyendo cuando el jugador hace coincidir la silueta con la ficha, de esta manera el elevador subirá cada vez más hasta alcanzar el final del tutorial.

### 7.3.3. Manejo y aprendizaje

Posteriormente a las mejoras del juego, se realizaron algunas pruebas referentes al aspecto lúdico del juego y su impacto a pequeña escala con los niños del Hogar de Bienestar Familiar mencionado, con tal propósito, se valoró el conocimiento de los pequeños en los temas referentes a este trabajo antes y después de jugar. Los resultados se presentan en la **Tabla 8** y la **Tabla 9**.

**Tabla 8**

*Valoración de conocimientos antes de usar la aplicación.*

Nombre	Edad	Colores conocidos	Tamaños conocidos	Figuras conocidas	Direcciones conocidas
José Manuel	4	3	1	3	I, D
Gabriela Gonzales	4	3	1	0	Ar, Ab
Cristian Moncayo	4	5	3	2	Ar
Samuel López	3	2	2	0	0
Juan Manuel	3	0	0	0	0
Yesica Alejandra	3	0	0	0	0
Emanuel Urbano	2	0	0	0	0

Convenciones para las direcciones:

- I: izquierda
- D: derecha
- Ar: arriba
- Ab: abajo
- ArI: arriba a la izquierda
- ArD: arriba a la derecha
- AbZ: abajo a la izquierda
- AbD: abajo a la derecha

**Tabla 9***Valoración de conocimientos después de usar la aplicación.*

Nombre	Edad	Colores conocidos	Tamaños conocidos	Figuras conocidas	Direcciones conocidas	Patrones resueltos
José Manuel	4	5	3	5	Todas	todos
Gabriela Gonzales	4	6	3	5	I, D, Ar, Ab, ArD	todos
Cristian Moncayo	4	6	3	3	I, D, Ar, Ab, ArD, ArI	todos
Samuel López	3	2	2	0	I, D	0
Juan Manuel	3	0	0	0	0	0
Yesica Alejandra	3	0	0	0	0	0
Emanuel Urbano	2	0	0	0	0	0

Como se ve en las tablas, los niños de cuatro años presentaron avances significativos en sus conocimientos, mientras que los de menor edad tuvieron un aprendizaje casi nulo, y por esta razón es necesario que el rango de edad, primeramente de tres a seis años fuera modificado empezando por los cuatro años como edad mínima, esto se debió a causa de que los más pequeños tenían muy poca atención y eran muy dispersos por lo que solo se distraían un rato con el juego y luego estaban más pendientes de lo que estaban haciendo sus demás compañeros; algunos de estos jugaban y asociaban pero no memorizaban lo que se les enseñaba a través del juego, por esta razón es necesario aclarar que el juego o la aplicación, es solo una herramienta para que la persona que se toma el tiempo de jugar con el niño pueda enseñarle los temas que aquí se tratan mientras el pequeño se divierte.

Respecto al juego de los patrones, los niños de cuatro años no tuvieron mayores dificultades para resolverlos mientras se les orientara y se les ayudara a identificar la simetría presente, por lo que aunque al primer intento no todos lo lograron, debido a que el patrón se redibuja cuando la respuesta es errada, al final siempre podían superar el reto, aunque algunos con más dificultad que otros, esto se justifica por el hecho de que los patrones no son de gran complejidad aunque ésta va en aumento.

Cabe resaltar que el juego en general posee varios aspectos lúdicos más allá de los mencionados, en especial en lo referente a motricidad fina, coordinación mano ojo, percepción plana y algunos otros aspectos presentes en la temprana edad, pero para este trabajo se omiten hasta este punto estos temas que son de índole pedagógica más profunda y se dejan como proyectos de estudio a futuro en colaboración con expertos en el tema de la psicomotricidad en la temprana edad.



**Figura 58.** Público infantil del Hogar los Jilgueros.

### 7.3.4. Implementación del Tangram

Aunque este juego no forma parte de los objetivos de este trabajo, se presentan algunos resultados de implementar este juego en algunos niños más grandes. Quienes probaron este juego fueron:

**Tabla 10**

*Resultados de implementación del tangram.*

Nombre	Edad	Siluetas resueltas	Ayuda
Cristian Fernando Realpe	11	7	si
Juan Esteban García	10	4	si
Samuel Alexander Realpe	8	17	si

De manera general, estos y otros niños que no se documentan aquí, lograron resolver algunas siluetas del juego, aunque los que aquí se mencionan necesitaron, algo de ayuda para empezar, es decir que se les dejaba armar libremente la figura, pero cuando llegaban a un punto en el que se estancaban, se les ofrecía la ayuda de ubicar una ficha correctamente para pudiera continuar, algo de esto se puede ver al final del video promocional del juego (disponible en: <http://bit.ly/2tR0uOs>).

El niño que logró armar la mayor cantidad de siluetas, lo logró con muy poca ayuda; solo se le ayudo en las primeras figuras, luego pudo asimilar la lógica del juego por sí mismo. El que armó menos siluetas necesitó un poco más de ayuda además de que su madre declaró que posee problemas de atención. Una posible razón de por qué el niño de menor edad pudo resolver mejor el juego, puede ser por el hecho de que, al ser menor, es más fácil llamar su atención, mientras que los niños más grandes suelen ser más exigentes y necesitan un nivel de emoción más grande para poder captar su atención.



## 8. CONCLUSIONES

- Se desarrolló un software capaz de reconocer y analizar figuras geométricas, gracias al uso de algoritmos de reducción de un contorno a una serie de puntos básicos como el algoritmo de Douglas-Peucker complementado con descriptores como el factor de forma entre otros, lo que hizo la tarea más factible y eficiente.
- En el procesamiento de video, es de gran ayuda disminuir la resolución del fotograma, pero teniendo en cuenta que esto genera una pérdida en la información que dificulta el reconocimiento, por lo que es necesario encontrar el punto de equilibrio en que la pérdida es mínima y el rendimiento sea el máximo posible.
- Emplear un cámara de parámetros no configurables para el procesamiento de video, puede generar sobrecarga en el proceso debido a la compensación de luz y retardo en la rata de fotogramas que el software del dispositivo configura automáticamente en base a las condiciones de iluminación.
- Se diseñó e implementó los juegos para las lúdicas, y para esto ha sido necesario utilizar colores llamativos y figuras que sirvan para asociarse a algún tema pedagógico que se esté tratando de transmitir.
- En el aprendizaje respecto a los temas aquí tratados, el que es de mayor dificultad para los niños, es la memorización los nombres de las figuras geométricas, en especial las que poseen mayor cantidad de lados.
- El aprendizaje de los tamaños relativos fue el más fácil de conseguir, dada su simplicidad todos los niños de cuatro años los aprendieron.
- El aprendizaje de los colores es más simple para los pequeños si estos se asocian con elementos ya conocidos para ellos.
- Los niños de cuatro años aquí tratados pueden identificar patrones simples, siempre que se les ayude a ver la simetría que presentan.
- Se creó una interfaz amigable con el público infantil en el rango establecido, teniendo en cuenta que es más atractiva si se emplean animaciones como las usadas para smartphones, dado el atractivo que estos dispositivos tienen para con los niños.
- Es posible crear un juego lúdico que enlace las temáticas básicas de un videojuego en 2D y el procesamiento digital de imágenes.

## 9. RECOMENDACIONES

- En el procesamiento de video es mejor si se utilice una cámara de parámetros configurables, específicamente la velocidad de fotogramas o *framerate*, esto evitará sobrecargas en el procesamiento cuando la intensidad en la iluminación disminuya.
- Para proyectos futuros, se aconseja utilizar elementos de la vida diaria en el juego que ayuden a los niños a memorizar los colores, tamaños y figuras, además de continuar probando el impacto del software para ampliar los horizontes de la aplicación, no solo para crear juegos lúdicos para niños pequeños, sino que seguir desarrollando lúdicas más complejas para temas más avanzados como los tocados en mecánica clásica, teoría de circuitos y otras áreas de las matemáticas a nivel escolar, también se recomienda que se mida el impacto del juego en lo referente a motricidad fina y coordinación entre otros aspectos presentes en el desarrollo cognoscitivo y motriz en los niños de cuatro a seis años, en colaboración con expertos en el tema.
- El uso del lenguaje QML es de gran ayuda para programar gráficos, puesto que es relativamente fácil de aprender y muy útil para crear aplicaciones y juegos con las estéticas modernas, además de que se puede relacionar con los lenguajes C++ y JavaScript, lo que facilita ligar el procesamiento de imágenes con aplicaciones visualmente atractivas.

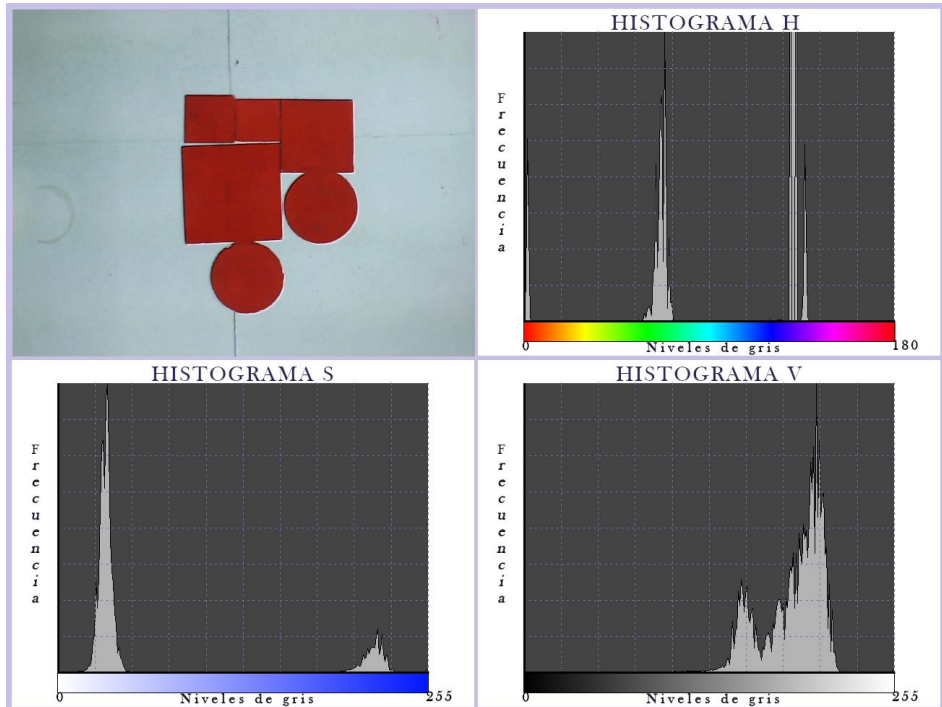
## BIBLIOGRAFÍA

- Aguilera, L. (16 de Noviembre de 2015). *PAI (Psico Ayuda Infantil)*. Obtenido de PAI (Psico Ayuda Infantil): <http://www.psicoayudainfantil.com/los-beneficios-del-tangram/>
- Cadillo León, J. R. (08 de Marzo de 2012). *Educaión, Sistemas, Redes y TIC*. Obtenido de Educaión, Sistemas, Redes y TIC: <https://conocimientoysistemas.wordpress.com/tag/vision-artificial/page/2/>
- Contreras, P. (14 de diciembre de 2013). *WIKIMEDIA COMMONS*. Obtenido de WIKIMEDIA COMMONS: [https://commons.wikimedia.org/wiki/File:Sensor\\_de\\_imagen\\_de\\_microscopio\\_usb\\_tipo\\_webcam.JPG](https://commons.wikimedia.org/wiki/File:Sensor_de_imagen_de_microscopio_usb_tipo_webcam.JPG)
- Fabiola, R. (18 de diciembre de 2015). *educ.ar*. Recuperado el 25 de octubre de 2018, de educ.ar: <https://www.educ.ar/noticias/129250/las-tic-en-la-educacion-inicial>
- Faraón Llorens-Largo, F. J.-D.-A.-R.-C.-C. (Marso de 2016). Gamificación del Proceso de Aprendizaje: Lecciones Aprendidas. *VAEP-RITA*, 4(1), 1.
- Fernández, J. L. (1 de abril de 2013). *FISICALAB*. Obtenido de FISICALAB: <https://www.fiscalab.com/apartado/camara-fotos#contenidos>
- Gary Bradski, A. K. (2008). *Learning OpenCV*. (M. Loukides, Ed.) Sebastopol, California, Estados Unidos: O'Reilly. Recuperado el 13 de Octubre de 2018, de <https://www.bogotobogo.com/cplusplus/files/OReilly%20Learning%20OpenCV.pdf>
- Koning, E. d. (25 de junio de 2011). *CODE PROJECT*. (C. Maunder, Ed.) Recuperado el 15 de octubre de 2018, de CODE PROJECT: [https://www.codeproject.com/Articles/114797/Polyline-Simplification#\\_articleTop](https://www.codeproject.com/Articles/114797/Polyline-Simplification#_articleTop)
- Maureen Meneses Montero, M. d. (septiembre de 2001). EL JUEGO EN LOS NIÑOS: ENFOQUE TEÓRICO. *EDUCACIÓN*, 25(2), 5-7. Recuperado el 25 de octubre de 2018, de <http://www.redalyc.org/pdf/440/44025210.pdf>
- MrSpyDoS. (3 de enero de 2012). *WIKIMEDIA COMMONS*. Obtenido de WIKIMEDIA COMMONS: [https://commons.wikimedia.org/wiki/File:Photo\\_lenses\\_with\\_a\\_focal\\_length\\_and\\_angle.png](https://commons.wikimedia.org/wiki/File:Photo_lenses_with_a_focal_length_and_angle.png)
- Ovide Decroly, E. M. (2002). *El juego educativo: iniciación a la actividad intelectual y motriz*. Madrid: Ediciones Morata, S.L.
- Palacios, A. (12 de 03 de 2018). <https://viviendomontessori.com>. Obtenido de Viviendo Montessori: <https://viviendomontessori.com/metodo-montessori/>

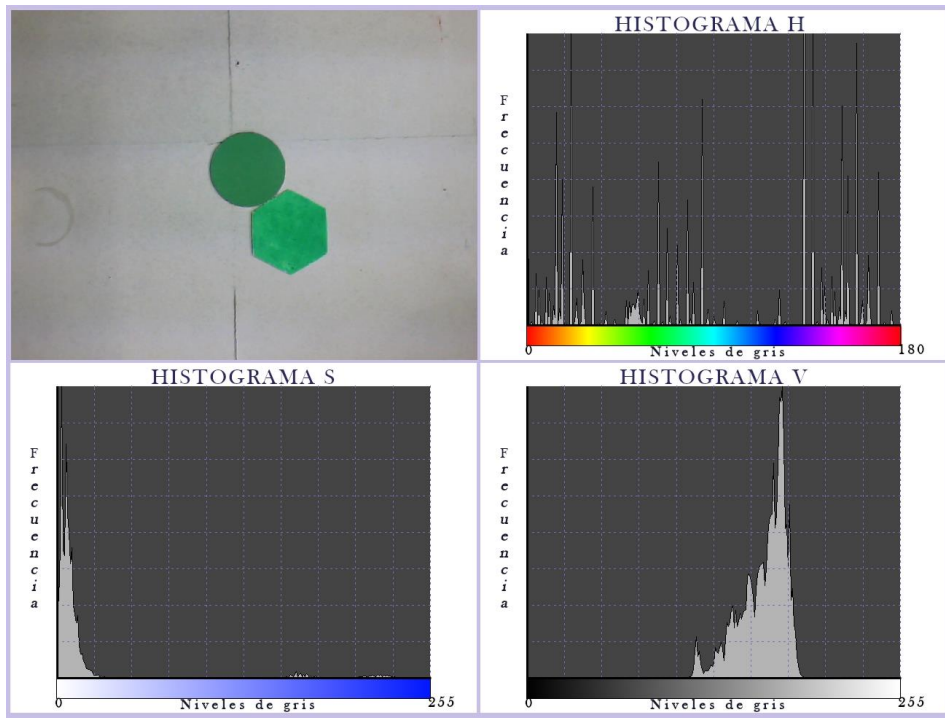
- Pediatrics, A. A. (01 de Noviembre de 2013). Children, Adolescents, and the Media. *PEDIATRICS*, 132(5), 958-960. Recuperado el 13 de 08 de 2017, de <http://pediatrics.aappublications.org/content/132/5/958>
- Pellissier, H. (2 de Septiembre de 2016). *Great! SCHOOLS.org*. Obtenido de Great! SCHOOLS.org: <https://www.greatschools.org/gk/articles/ingeniosas-maneras-de-ensenarle-patrones-a-su-prescolar/?lang=es>
- Roldán, M. J. (12 de 3 de 2018). *Guía Infantil*. Obtenido de <https://www.guiainfantil.com/articulos/bebes/desarrollo/que-aprende-el-bebe-con-3-anos/>
- Torres, A. D. (Abril de 1996). Procesamiento digital de imágenes. *Perfiles Educativos*(72), 2. Recuperado el 11 de Octubre de 2018, de <http://www.redalyc.org/pdf/132/13207206.pdf>
- Wainschenker, D. R. (2011). Procesamiento Digital de Imágenes. *Optativa Área Procesamiento de señales*, 4. Obtenido de <http://www.exa.unicen.edu.ar/catedras/pdi/FILES/TE/CP1.pdf>
- Welleman, P. (18 de enero de 2008). *WIKIMEDIA COMMONS*. Obtenido de WIKIMEDIA COMMONS: <https://commons.wikimedia.org/wiki/File:Chipincamera.jpg>
- Zurira A., O. (1992). *Enciclopedia del Test Razonamiento Abstracto*. Popayán, Colombia: Ediciones Colombia.

## ANEXOS

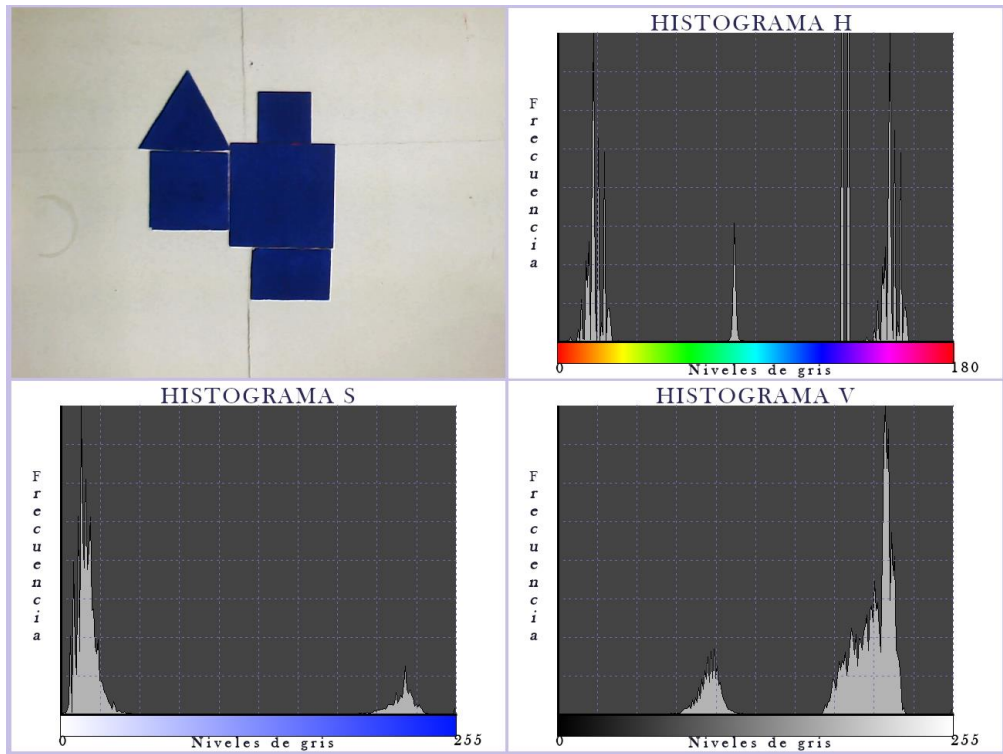
### Anexo A. Histogramas HSV en respuesta a los colores de interés.



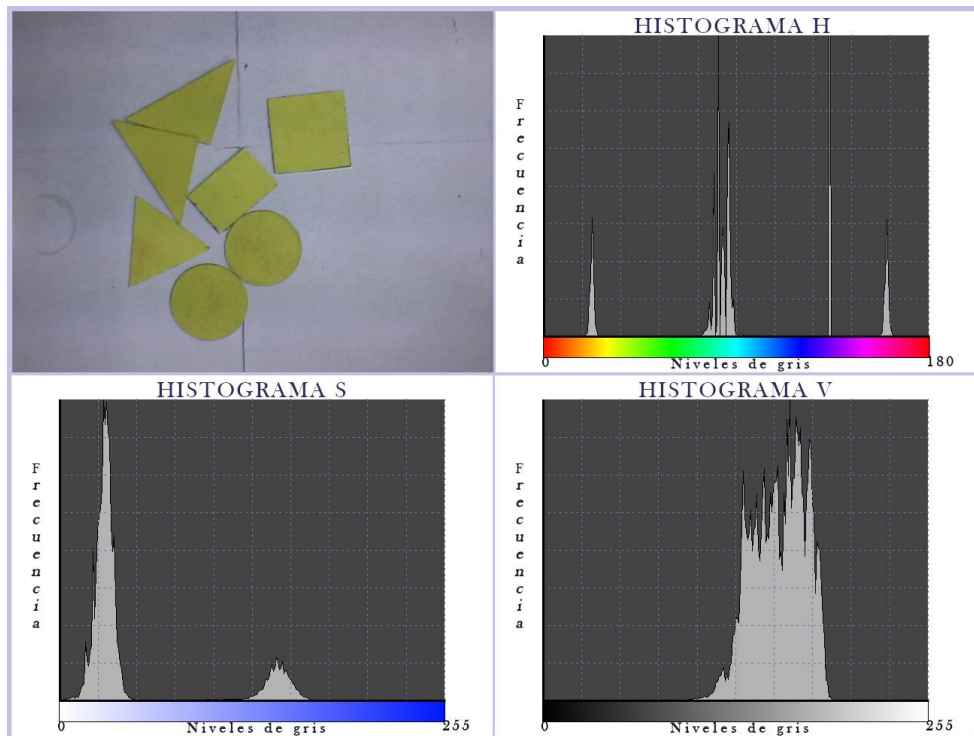
*Figura A.1. Histogramas para las fichas rojas.*



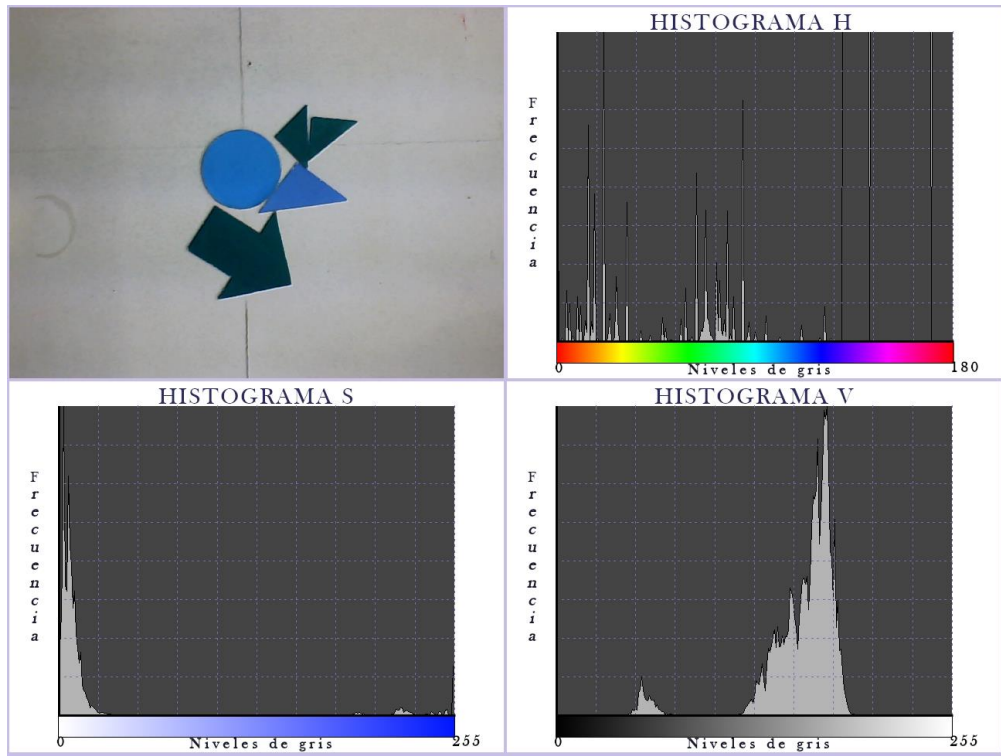
*Figura A.2. Histograma para las fichas verdes.*



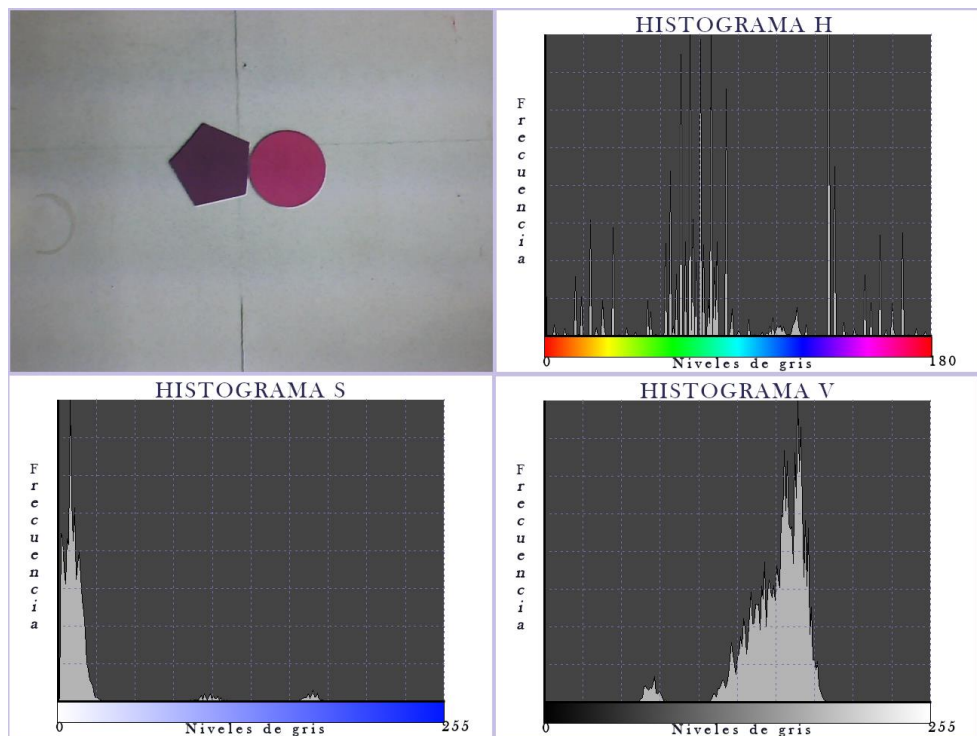
**Figura A.3.** Histograma para las fichas azules.



**Figura A.4.** Histograma para las fichas amarillas.



**Figura A.5.** Histograma para las fichas color cian.



**Figura A.5.** Histograma para las fichas color violeta.

## Anexo B. Estructura del programa para el PDI.

### Cabecera

Librerías generales  
Librerías auxiliares  
Librerías de OpenCV

### Declaraciones globales

Variables globales  
Subrutinas

### Programa principal

```
int main (int argc, char *argv[])
{

    Declaración de variables locales
    Instanciar la clase del receptor

    Bucle infinito
    while(true){
        abrir cámara
        capturar fotograma
        establecer tamaño de imagen
        aplicar filtro de mediana
        crear una imagen en HSV
        configurar interrupción por mouse
        promediar los valores obtenidos mediante la interrupción
        segmentar en base a los valores anteriores
        detectar contornos con findContours()
        limpiar receptor
        analizar los contornos
        for (size_t i = 0; i < contours.size(); i++){
            filtrar los contornos por tamaño
                if(contourArea(contours[i])>500)

                    detección del color con subrutina
                    detección de la figura con subrutina
                    crear rectángulo que contiene al contorno
                    calcular centro geométrico
                    calcular tamaño mediante el área
                    calcular posición en la grilla
                    limpiar vector de datos

                    if (figure=="triangulo"){
                        poner etiquetas
                        llenar el vector de datos
                        entregar información al receptor
                    }
                    else if (figure=="cuadrado"){
                        poner etiquetas
                        llenar el vector de datos
                        entregar información al receptor
                    }
                    else if (figure=="rectangulo"){
                        poner etiquetas
                        llenar el vector de datos
                        entregar información al receptor
                    }
                }
            }
        }
    }
```



```

else if (figure=="paralelogramo"){
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f1"){ Flecha derecha
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f2"){ Flecha diagonal derecha
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f3"){ Flecha arriba
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f4"){ Flecha diagonal izquierda
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f5"){ Flecha izquierda
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f6"){ Flecha izquierda abajo
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f7"){ Flecha abajo
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="f8"){ Flecha derecha abajo
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
else if (figure=="circulo"){
    poner etiquetas
    llenar el vector de datos
    entregar información al receptor
}
}
dibujar contornos
}
}

```

#### **modo tangram**

```

if(receiver.tangramModeActive==true){

    redimensionar fotografias

```

```

        calibrar cámara si está disponible
        abrir silueta en cámara
        restar silueta y segmentación para
        calcular error
        evaluar juego
    }
    Dibujar grilla
    Dibujar patrón si está disponible
    Mostrar fotogramas
    return 0;
}

Subrutinas

void interrupcion( int event, int x, int y, int,void*)
{
    Tomar valores de los pixeles y guardarlos en un vector
}

static double angulo(cv::Point pt1, cv::Point pt2, cv::Point pt0)
{
    Calcula el Angulo entre tres puntos
}

string NivelColor(Mat contorno){
    determina el color a partir del plano H
    int h=vecth[20]; // este valor es el dato central (mediana) plano H
    string color;
    // cout <<"h: "<<h<<endl;
    if(h<=5 || h>174)
        color="Rojo";
    else if (h<=15 && h>5)
        color="Naranja";
    else if (h<=35 && h>15)
        color="Amarillo";
    else if (h<=88 && h>35)
        color="Verde";
    else if (h<=109 && h>88)
        color="Cian";
    else if (h<=123 && h>109)
        color="Azul";
    else if (h<=174 && h>123)
        color="Violeta";

    return(color);
}

Mat filter(Mat img){
    Filtro morfológico
    Apertura
    cierre
}

void drawGrid() {
    dibuja la grilla
}

```

```

void drawPattern(int nf,int patternType){
    crea y dibuja los patrones
}

int indexNum(QPoint p){
    determina la posición en la grilla
}

Función para determinar la figura del contorno
string figuraContorno(Mat contorno){

    string fig;
    vector<Point> approx;
    reducir el número de puntos usando del algoritmo de Douglas-
    Peucker
    approxPolyDP(contorno,approx,5,true);
    double area = cv::contourArea(contorno);
    perímetro
    double p=arcLength(contorno,1);
    factor de forma
    double ff = 40*CV_PI*area/(p*p);

    if (approx.size() == 3)
    {
        fig="triangulo";
    }
    else if (approx.size() >= 4 && approx.size() <= 6)
    {
        // Obtener el numero de vertices
        int vtc = approx.size();

        // Obtener todos los ángulos (cosenos)
        std::vector<double> cos;
        for (int j = 2; j < vtc + 1; j++)
        cos.push_back(angle(approx[j%vtc], approx[j - 2], approx[j -
        1]));

        //Ordenar los ángulos obtenidos
        std::sort(cos.begin(), cos.end());

        // Obtener los ángulos máximos y mínimos
        double mincos = cos.front();
        double maxcos = cos.back();

        //Obtener los ángulos y determinar el tipo de figura según los
        //ángulos obtenidos y el número de vértices
        if (vtc == 4){
            if(mincos >= -0.1 && maxcos <= 0.3){//ángulo sercano a 90°
                Rect r = cv::boundingRect(contorno);

                if (std::abs(1 - ((double)r.width / r.height)) <= 0.1){
                    fig="cuadrado";
                }
                else{
                    fig="rectángulo";
                }
            }
        }
        else{

```

```

        fig="paralelogramo";
    }
}
else if (vtc == 5)
    fig="pentágono";
}
else if (vtc == 6)
    fig="hexágono";
}
}
else if (ff>=8.4){fig="círculo";}
else if (approx.size() == 7){
    // Obtener el número de vértices para la flecha
    int vtc = approx.size();
    double ang,ang1,d,d1;
    int m,n,m1,n1,x,y,x1,y1;

    bucle para determinar el ángulo de la flecha
    for (int jj = 0; jj < 7 ; jj++){
        m=jj+1; n=jj+3; m1=jj-1; n1=jj-3;
        if(m>=vtc){m=m-vtc;}
        if(n>=vtc){n=n-vtc;}
        if(m1<0){m1=m1+vtc;}
        if(n1<0){n1=n1+vtc;}
        x=(approx[m].x+approx[m1].x)/2;
        y=(approx[m].y+approx[m1].y)/2;
        x1=(approx[n].x+approx[n1].x)/2;
        y1=(approx[n].y+approx[n1].y)/2;
        d=sqrt((approx[m].x-approx[m1].x)*(approx[m].x-
approx[m1].x)+(approx[m].y-approx[m1].y)*(approx[m].y-approx[m1].y));
        d1=sqrt((approx[n].x-approx[n1].x)*(approx[n].x-
approx[n1].x)+(approx[n].y-approx[n1].y)*(approx[n].y-approx[n1].y));

        ang=acos(angle(approx[jj],Point(x1,y1),Point(x,y)))*180/CV_PI;
        //
        if(ang<=190 && ang>=170 && d>(d1*1.5)){

        ang1=acos(angle(Point(640,y1),approx[jj],Point(x1,y1)))*180/CV_PI;

            if(approx[jj].y>y1){ang1=360-ang1;}
            jj=7;
        }
    }

    if(ang1<=15 || ang1>=345){fig="f1";}
    else if (ang1<=75 && ang1>15){fig="f2";}
    else if (ang1<=105 && ang1>75){fig="f3";}
    else if (ang1<=165 && ang1>105){fig="f4";}
    else if (ang1<=195 && ang1>165){fig="f5";}
    else if (ang1<=255 && ang1>195){fig="f6";}
    else if (ang1<=285 && ang1>255){fig="f7";}
    else if (ang1<345 && ang1>285){fig="f8";}
    else{fig="flecha";}
}
return(fig);
}
}

```

## Anexo C

### LA NEUROEDUCACIÓN

Se presenta en este anexo un enfoque a el juego educativo desde el punto de vista de la neurociencia bajo la nueva corriente conocida como la neuroeducación, la cual aprovecha los conocimientos referentes a los procesos cerebrales que influyen en el aprendizaje para mejorar las técnicas de enseñanza y desarrollar nuevos métodos de educación que refuercen los procesos cognoscitivos, en especial en lo referente a la educación de los niños.

#### **Autores relevantes**

En la actualidad, se cuenta con multitud de publicaciones y estudios en neuroeducación y neurodidáctica. Autores como Francisco Mora, José Antonio Marina, Ignacio Morgado, Eric Kandel, Judy Willis, Paul Howard-Jones, Eric Jensen, Charo Rueda, Tomás Ortiz, José Ramón Alonso, Uta Frith, SarahJayne Blakemore, Rosa Casafont, Marta Ligioiz, José Ramón Gamo, Chema Lázaro y Jesús C. Guillén, Jaime Romano, entre otros se destacan en los últimos 10 años<sup>1</sup>. En este breve enfoque solo se hará mención de algunos de ellos.

#### **Generalidades**

De manera general, la neuroeducación señala que la emoción, la sorpresa, la experimentación y el deporte son ingredientes necesarios para sumar conocimientos<sup>2</sup>, desde este punto de vista es innegable que es necesario promover nuevas formas de aprendizaje que rompan el esquema tradicional, en la búsqueda de generar estos ingredientes fundamentales durante el proceso de enseñanza en los niños, teniendo como objetivo principal el lograr captar la atención del alumnado, en este punto es donde la tectología o mejor dicho las TIC toman un rol contundente, ya que está claro que este tipo de dispositivos logran llamar la atención de los infantes con mucha facilidad, esto recalca la necesidad de emplear las nuevas tecnologías en los nuevos métodos de

---

<sup>1</sup> Bueno, D., & Forés, A. (2008). 5 principios de la neuroeducación que la familia debería saber y poner en práctica. *Revista Iberoamericana de Educación*, 78, 14. Obtenido de <https://rieoei.org/RIE/issue/download/282/vol.%2078%2C%20n%C3%BAm.%201>

<sup>2</sup> Sáez, C. (2014). NEUROEDUCACIÓN, CÓMO EDUCAR CON CEREBRO. *Quo*, 75-78. Obtenido de [http://www.ub.edu/geneticaclass/davidbueno/Articles\\_de\\_divulgacio\\_i\\_opinio/Altr/es/Neuroeducacion-QUO.pdf](http://www.ub.edu/geneticaclass/davidbueno/Articles_de_divulgacio_i_opinio/Altr/es/Neuroeducacion-QUO.pdf)

enseñanza, haciendo énfasis en el impacto emocional que se puede generar en el infante, puesto que según la neurociencia, es la emoción la clave para un buen aprendizaje<sup>3</sup>.

### **La emoción como factor fundamental**

Francisco Mora, neurocientífico, autor de *Neuroeducación. Solo se puede aprender aquello que se ama*, uno de los primeros manuales sobre neuroeducación, señala que el binomio emoción-cognición es indisoluble, intrínseco al diseño anatómico y funcional del cerebro. Señala Sáez que, al parecer, la información que captamos por medio de los sentidos pasa por el sistema límbico o cerebro emocional antes de ser enviada a la corteza cerebral, encargada de los procesos cognitivos. Dentro del sistema límbico, la amígdala tiene una función esencial. Es una de las partes más primitivas del cerebro y se activa ante eventos que considera importantes para la supervivencia, lo que consolida un recuerdo de manera más eficiente<sup>4</sup>. En este contexto se rescata que cualquier herramienta o recurso capaz de generar emoción en los niños puede ser empleado para transmitir conocimiento de manera eficiente y duradera; un videojuego es un ejemplo claro de cómo llamar la atención de los más pequeños, por lo que es común que se implementen diversos diseños de éstos para mejorar el aprendizaje en el estudiantado.

Jaime Romano, neurólogo mexicano, investiga el cerebro desde hace más de 30 años, explica que la información que entra por los sentidos, que posteriormente se procesa en el cerebro para convertirse en aprendizaje, tiene que ver con procesos emocionales, en base a esto, Romano trabaja en el diseño de videojuegos orientados a distintas edades. “Queremos mejorar la capacidad emocional y mental de los estudiantes, los procesos de cálculo, de comprensión, y eso repercutirá en que aprenderán mejor las matemáticas, a leer y a entender los textos, a fijar su atención”.<sup>5</sup>

---

<sup>3</sup> Campos, A. L. (2010). NEUROEDUCACIÓN: UNIENDO LAS NEUROCIENCIAS Y LA EDUCACIÓN EN LA BÚSQUEDA DEL DESARROLLO HUMANO. *la educ@ción*(143), 6. Obtenido de [http://www.educoea.org/portal/La\\_Educacion\\_Digital/laeducacion\\_143/articulos/neuroeducacion.pdf](http://www.educoea.org/portal/La_Educacion_Digital/laeducacion_143/articulos/neuroeducacion.pdf)

<sup>4</sup> Obtenido del mismo artículo referenciado en 1.

<sup>5</sup> Obtenido del mismo artículo referenciado en 1.

## **El factor sorpresa**

Describe Sáez que la sorpresa es otro factor que activa la amígdala. El cerebro es un órgano al que le gusta procesar patrones, entender cosas que se repiten siempre de la misma forma, es la manera como se enfrenta al mundo que lo rodea. Ahora bien, todo aquello que no forma parte de esos patrones se guarda de manera más profunda en el cerebro. De ahí que usar en la clase elementos que rompan con la monotonía benefician el estudio<sup>6</sup>. En esta parte es donde se destaca el empleo de videojuegos y las nuevas técnicas de gamificación para mejorar la memorización de conceptos en el ámbito educativo, puesto que nos son herramientas comunes dentro de la atmósfera educativa clásica, se destaca también que la adopción de mecánicas lúdicas basadas en entretenimiento digital, puede proporcionar valiosos recursos que brinden soporte al estudiante a seguir procesos particulares, usar guías para el autoconocimiento, tomar decisiones vocacionales, mejorar su autonomía y continuar sus experiencias de aprendizaje de manera diferente y divertida<sup>7</sup>.

## **Contextualización referente a este trabajo**

Bajo esta aproximación a la neuroeducación, se resalta que la herramienta desarrollada en este trabajo contiene elementos clave para fomentar el aprendizaje en los niños, dado que cuenta con el factor sorpresa por mostrar un nuevo método de enseñanza mediante un videojuego y la visión por computador, además de generar emoción en los niños al interactuar con la aplicación, su estética, ambiente musical y las fichas físicas que al conectarse con el videojuego mediante el procesamiento digital de imágenes, resulta ser como “magia” ante los ojos de los pequeños. En este contexto la motivación es claramente alcanzada.

---

<sup>6</sup> Obtenido del mismo artículo referenciado en 1.

<sup>7</sup> Ferrer, S. C., Reina, M. F., Daniela, N., Padrón, P., Montero, M. E., & Ferrer, E. E. (2008). La gamificación como herramienta en el trabajo docente del orientador: innovación en asesoramiento vocacional desde la neurodidáctica. *Revista Iberoamericana de Educación*, 177. Obtenido de <https://rieoei.org/RIE/issue/download/282/vol.%2078%2C%20n%C3%BAm.%201>