**UNIVERSIDAD NACIONAL DE INGENIERÍA**

Facultad de Ingeniería Industrial y de Sistemas

****

**Sistema de Reconocimiento de objetos de un salón de clases**

**CURSO:** Inteligencia Artificial Avanzada

**DOCENTE:** ZELA MORAYA, WESTER EDISON

**INTEGRANTES:**

Acuña Herrera, Walter Mario

Ccaico Intuscca, Luis Fernando

Rojas Llactahuamani, José Manuel

# 

# **CAPÍTULO 1.**

# **1. DESCRIPCIÓN DEL SISTEMA**

|  |  |
| --- | --- |
| Nombre del sistema | **Sistema de Reconocimiento de un salón de clases** |
| Descripción del Sistema | El sistema se divide en 5 fases:   1. **Convertir la imagen a escala de grises** Una vez que has cargado la imagen, lo primero que tienes que hacer es convertir a escala de grises con OpenCV. Cuando trabajamos con imágenes a color, el coste computacional crece de manera exponencial. 2. **Filtrar la imagen para eliminar el ruido** Las imágenes digitales no son perfectas. El ruido inherente de los propios dispositivos o los efectos contraproducentes por iluminación alteran la realidad. Es algo que no solo sucede en los entornos visuales. Cualquier señal digital o analógica está expuesta a diferentes fuentes de ruido. Debes vivir con ello y, sobre todo, debes saber cómo corregirlo. 3. **Aplicar el detector de bordes Canny** El proceso para detectar bordes con Canny se divide en 3 pasos.  * Detección de bordes con Sobel * Supresión de píxeles fuera del borde * Aplicar umbral por histéresis  1. **Buscar los contornos dentro de los bordes detectados** Hay que saber la diferencia que existe entre un borde y un contorno. Los bordes, como hemos visto anteriormente, son cambios de intensidad pronunciados. Sin embargo, un contorno es una curva de puntos sin huecos ni saltos es decir, tiene un principio y el final de la curva termina en ese principio. 2. **Dibujar dichos contornos** Por último nos queda dibujar los contornos que hemos encontrado en la imagen. Esto lo haremos a través de otro método de OpenCV.   Donde:   * imagen: es la imagen donde vamos a dibujar los contornos. * lista\_contornos: es la lista de Python con los contornos. * numero\_contornos: el número de contornos que queremos dibujar si son todos pasar -1. * color\_RGB: es un array o tupla con el color RGB del contorno. * grosor: grosor de la línea a dibujar. |
| Miembros del Equipo | Acuña Herrera Walter  Ccaico Instucca Luis Fernando  Rojas Llactahuamani José Manuel |

**2. DESCRIPCIÓN DEL PROBLEMA**

***No existe un método general con el que una máquina pueda lograr el reconocimiento de objetos imágenes de objetos que se encuentren dentro de un salón de clases.***

Actualmente existen diversos métodos que se utilizan para el reconocimiento de objetos, entre los cuales se mencionan:

**2.1. Métodos basados en apariencia**

Usa imágenes de ejemplo (llamadas plantillas o prototipos) de los objetos tomadas desde diferentes vistas.

Los objetos se ven diferentes bajo una variedad de condiciones:

* Cambios en la iluminación o color
* Cambios en la dirección de observación
* Cambios en tamaño / forma

**2.1.1. Emparejamiento de bordes**

* Usa técnicas de detección de bordes, tales como Algoritmo de Canny, para encontrar los bordes.
* Cambios en la iluminación y color usualmente no tiene mucho efecto en los bordes de la imagen
* Estrategia:

1. Detecta bordes en la plantilla y la imagen
2. Compara los bordes de la imagen para encontrar la plantilla
3. Debe considerar el rango de las posibles posiciones de la plantilla

* Medida:

1. Bueno – cuenta el número de bordes superpuestos. No es robusto a cambios en la forma
2. Mejor – Cuenta el número de pixeles del borde de la plantilla con alguna distancia de un borde en la imagen de búsqueda
3. Óptimo – Determina la distribución de probabilidad de la distancia para el borde próximo en la imagen de búsqueda (si la plantilla está en posición correcta). Estima la probabilidad de cada posición de la plantilla generando imagen

**2.1.2. Búsqueda Divide y Vencerás**

* Estrategia:
* Considera todas las posiciones como un conjunto (una celda en el espacio de las posiciones)
* Determina el salto menor en la puntuación en la mejor puntuación de la celda
* Si el salto es muy largo, poda la celda
* Si el salto no es muy largo divide la celda en subceldas y tratas cada subcelda recursivamente
* Paras cuando la celda sea “suficientemente pequeña”
* Esta técnica garantiza encontrar todas las parejas que cumplan el criterio (asumiendo que el menor salto es exacto)

**2.1.3. Emparejamiento por escala de grises**

* Las esquinas son (en su mayor parte) robusto a cambios en la iluminación, sin embargo desperdician mucha información
* Debes computar la distancia de pixel como una función de la posición del pixel y la intensidad del mismo
* Puede ser aplicado a imágenes en colores también

**2.1.4. Emparejamiento del gradiente**

* Otra forma de ser robusto a los cambios de la iluminación sin desperdiciar mucha información para comparar el gradiente de las imágenes
* El emparejamiento se realiza como el emparejamiento por escala de grises
* Alternativa simple: Usa la correlación (normalizada)

**2.2. Métodos basados en modelos**

En este tipo de métodos se parte de un modelo previo de cada uno de los objetos que se desean detectar. A partir de este modelo, se buscan aquellos métodos que sean capaces de encontrar puntos en la imagen y asociarlos a las primitivas (puntos, líneas, volúmenes) que definen el modelo, es decir, generar la información que permitirá diferenciar unos puntos de otros en función de las características que presenten dichos puntos. Una vez asociados, se deberá ser capaz de calcular la correspondencia entre el modelo y la imagen a analizar. A partir de este emparejamiento de puntos, el sistema debería ser capaz de obtener la posición y orientación del objeto.

Habitualmente estas técnicas requieren de imágenes de gran resolución. Cuando se utilizan estos sistemas, el algoritmo sabe con antelación el objeto que ha de representar y lo que intenta hacer es que corresponda la imagen del objeto con el modelo. El proceso que se suele seguir cuando se usan estas técnicas está formado por tres pasos:

* Construcción del modelo.
* Ajustar el modelo a la imagen de test.

Utilizar los parámetros del modelo ajustado para calcular la similitud entre la imagen de test y las imágenes de referencia para realizar el reconocimiento.

Los trabajos de detección basados en este método son muy extensa. Se puede realizar una posible división de todos estos trabajos en función del tipo de características que se buscan en la imagen y que se utilizan para generar los modelos A continuación se enumeran alguno de ellos:

**2.2.1. Métodos basados en detección de bordes.**

Los primeros trabajos de detección de objetos estaban basados en la detección de bordes debido a su eficiencia computacional y a su fácil implementación. Además, estos métodos son inmunes a los cambios de iluminación.

**2.2.2. Métodos basados en ajuste de plantilla.**

Estos métodos se basan en el ajuste de una plantilla bidimensional genérica cuando ésta se ve sometida a un conjunto de deformaciones afines. En general, el objetivo de este tipo de algoritmos consiste en encontrar, para una determinada imagen, los parámetros que definen la transformación que habría que aplicarle a la plantilla para obtener el mismo resultado que en la imagen. Una vez obtenida la plantilla es posible recuperar su pose mediante la homografía que define el plano que la forma y los parámetros intrínsecos de la cámara. Para realizar la detección, no se utiliza características locales del objeto sino que se toma toda la imagen correspondiente a la plantilla como fuente de información para la detección. En general, este tipo de métodos de detección presentan varias limitaciones: es poco robusta ante cambios de iluminación y posibles oclusiones en la plantilla.

**2.2.3. Métodos basados en extracción de puntos.**

Este tipo de métodos ya no se basa en marcas globales (como los contornos o las plantillas) sino que utiliza marcas naturales localizadas. Se basan en la detección de puntos individuales, lo que permite que dichos algoritmos sean inmunes a cambios de iluminación, oclusiones o errores de correspondencia. Normalmente, para obtener los puntos, este tipo de enfoque se basa en la información contenida en el plano imagen más que en la apariencia real del objeto. Los métodos de detección basados en extracción de puntos deben resolver fundamentalmente dos problemas:

**• Identificación de Puntos de Interés.**

En general, para identificar un punto se utiliza la información aportada por la textura de una pequeña región de la imagen situada en torno al punto. No todos los puntos proyectados del objeto son susceptibles a ser detectados en otras imágenes, pues la textura que rodea a un punto puede sufrir deformaciones proyectivas, cambios de iluminación y ruido. Por tanto, el proceso de identificación consiste en detectar los puntos proyectados del objeto que presenten ciertas características que los convierte en candidatos idóneos para ser detectados de nuevo en otras imágenes. Estos puntos se denominan Puntos de Interés.

**• Correspondencia entre imágenes.**

Para poder detectar el objeto y obtener la transformación que ha sufrido, es necesario un método de correspondencia entre puntos de interés extraídos de una imagen y los puntos de interés del objeto. Normalmente el método de correspondencia tiene asociado algún algoritmo de estimación robusta que permita descartar falsas correspondencias.

**2.2.4. Métodos basados en descriptores invariantes**

El concepto de estos métodos es el mismo que para el caso anterior. La diferencia es que cada punto de interés tendrá asociado un descriptor local. Dicho descriptor busca extraer información de un área localizada de la imagen (como puede ser la textura alrededor del punto de interés) y define un método de codificación que sea invariante a cambios de iluminación y transformaciones afines. Los descriptores deben ser definidos de tal modo que a su vez permitan separar de manera apreciable dos áreas que no pertenezcan al mismo punto.

**3. ALCANCE DEL SISTEMA**

El sistema que proponemos, Sistema de Reconocimiento de un salón de clases, podrá reconocer objetos que puedan existir en un salón de clases, **el reconocimiento de estos objetos será uno a la vez**, esto lo podrá lograr mediante una visión artificial específicamente con el reconocimiento de imágenes, para este caso proponemos el uso del aprendizaje profundo como modelo que pueda lograr el objetivo de reconocer los objetos.

Para lograr esto nos valemos de OpenCV, una libreria orientada a la visión artificial, esta libreria contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos (reconocimiento facial), calibración de cámaras, visión estérea y visión robótica. Aparte de esto OpenCV tiene una licencia BSD, lo que permite utilizar y modificar el código, tiene una comunidad de más de 47000 personas y más de 7 millones de descargadas. Es una librería muy usada a nivel comercial, desde Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota, etc.

Como entorno de lenguaje de programación usaremos Python, este lenguaje de programación es multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, [programación imperativa](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n#Clasificaci%C3%B3n_por_paradigmas) y programación funcional.

Aparte de estas caracteristicas Python abarca grandes librerias que nos facilita su uso, por ejemplo OpenCV, numpy, Matplotlib, Seaborn, etc.

1. Selección de la técnica para resolver el problema

**Detección de bordes**

La detección de bordes es una técnica muy utilizada que nos permite aislar los objetos y separarlos del fondo. Una vez obtenido los bordes, lo único que nos faltaría es detectar los diferentes contornos para poder contar los objetos. La técnica consta de cinco etapas. Cada una de estas se menciona a continuación:

**Preprocesamiento de la imagen**

En esta etapa una imagen monocromática proveniente de un sensor CCD es primeramente binarizada manualmente.

**Determinación del número de conglomerados.**

Se obtiene a través del etiquetado de todas y cada una de las componentes conectadas en la imagen binarizada. La imagen es barrida de izquierda a derecha y de arriba a abajo hasta localizar el primer píxel negro, asignándole la primera etiqueta. Una vez encontrado este píxel, sus ocho vecinos son analizados, si cualquiera de éstos es negro es etiquetada con la misma etiqueta del píxel central.

**Determinación del número de hoyos en una imagen.**

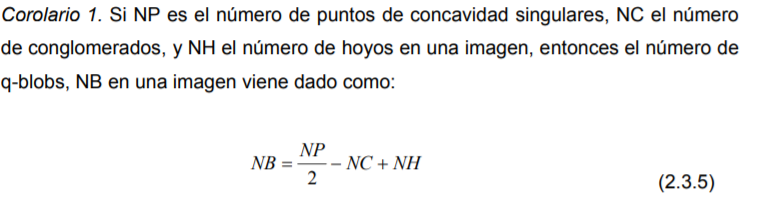
Consiste en invertir la imagen original, contar el número de regiones usando el procedimiento para contar conglomerados, para obtener el valor deseado.

**Determinación del número de puntos de concavidad singulares.**

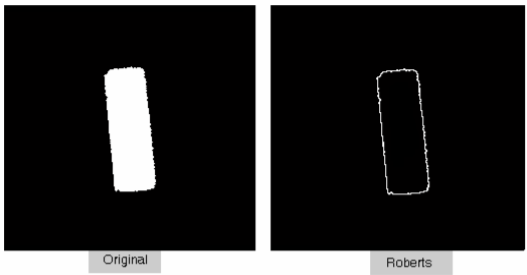
La imagen original es primeramente dilatada usando un elemento estructural circular de diámetro siete. La imagen resultante es enseguida erosionada con el mismo elemento estructural. Esto da como resultado una imagen cerrada, morfológicamente hablando. La imagen original es restada de esta imagen cerrada para obtener la imagen final, la cual contendrá tantas regiones conexas como puntos de concavidad singulares haya en la imagen original.

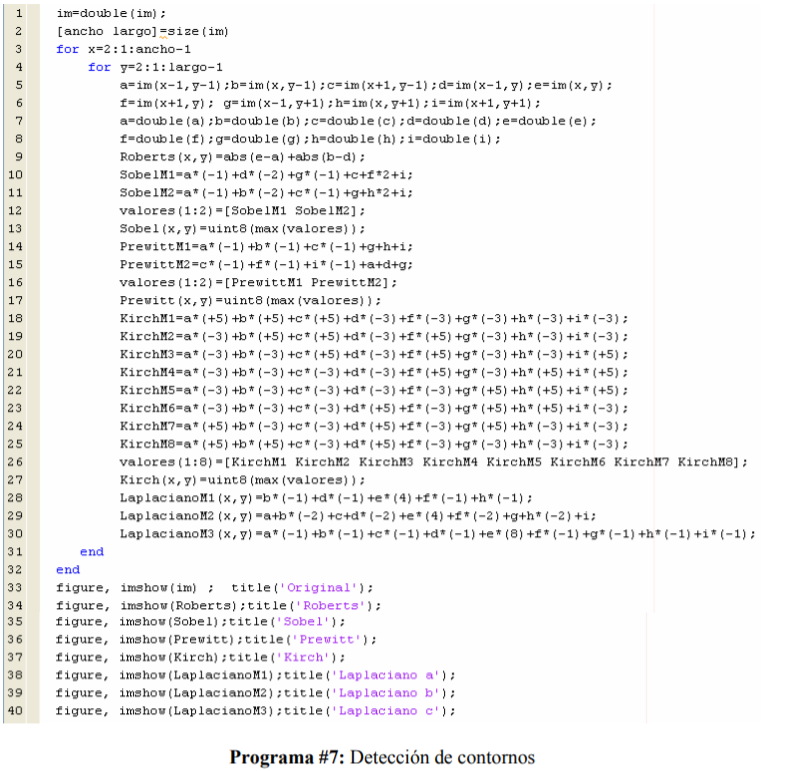
**Determinación del número de q-blobs en la imagen**

El cálculo del número de q-blobs se reduce a aplicar lo siguiente.



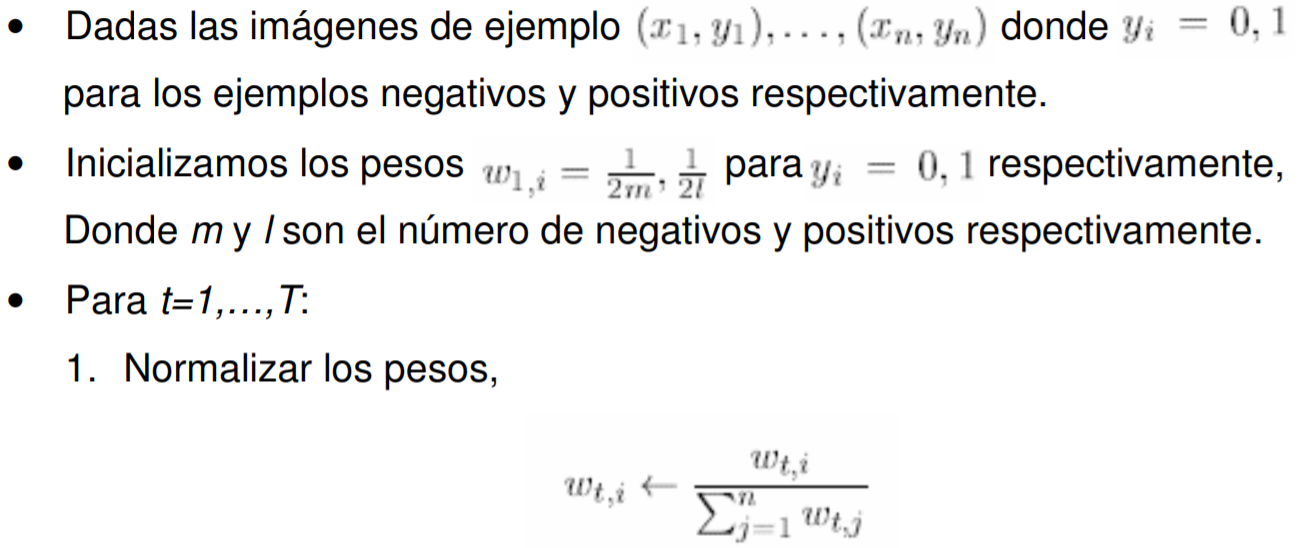
**Algoritmo de detección de contorno**

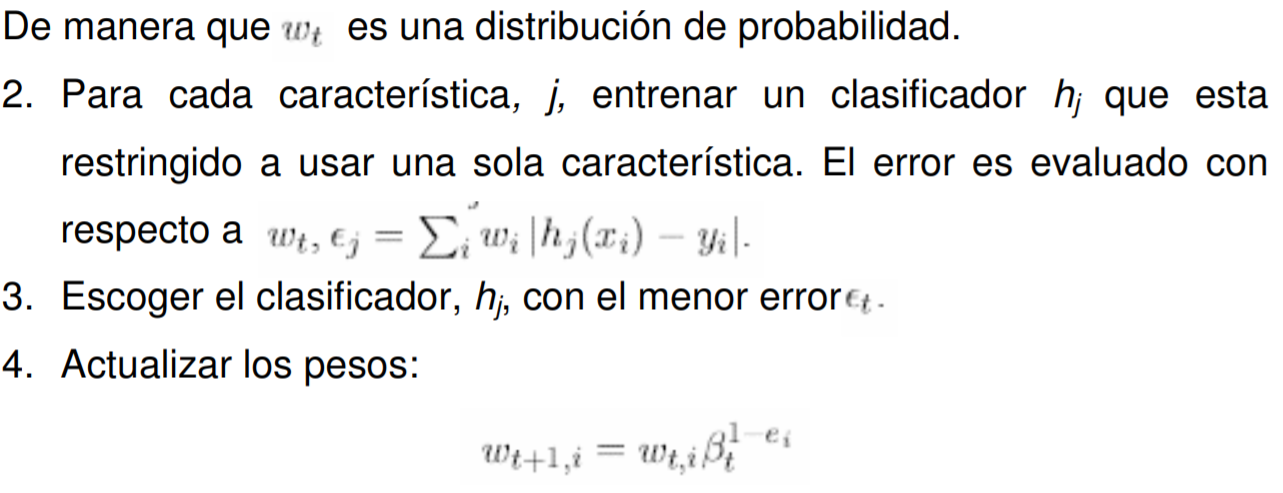


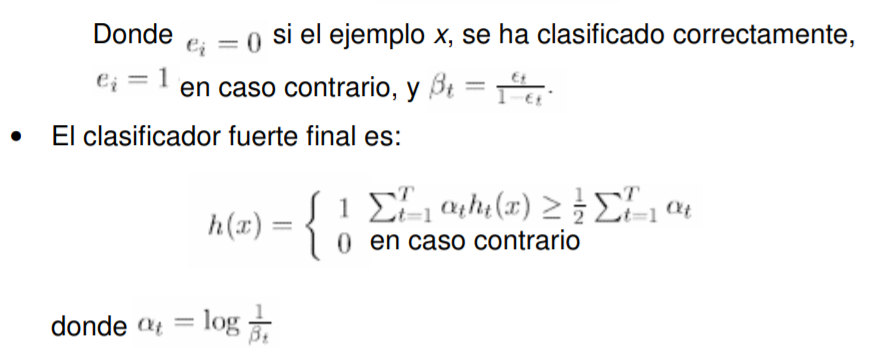


**Algoritmo de aprendizaje del clasificador**

El clasificador está basado en una cascada de clasificadores binarios entrenados mediante el algoritmo de Adaboost descrito a continuación.





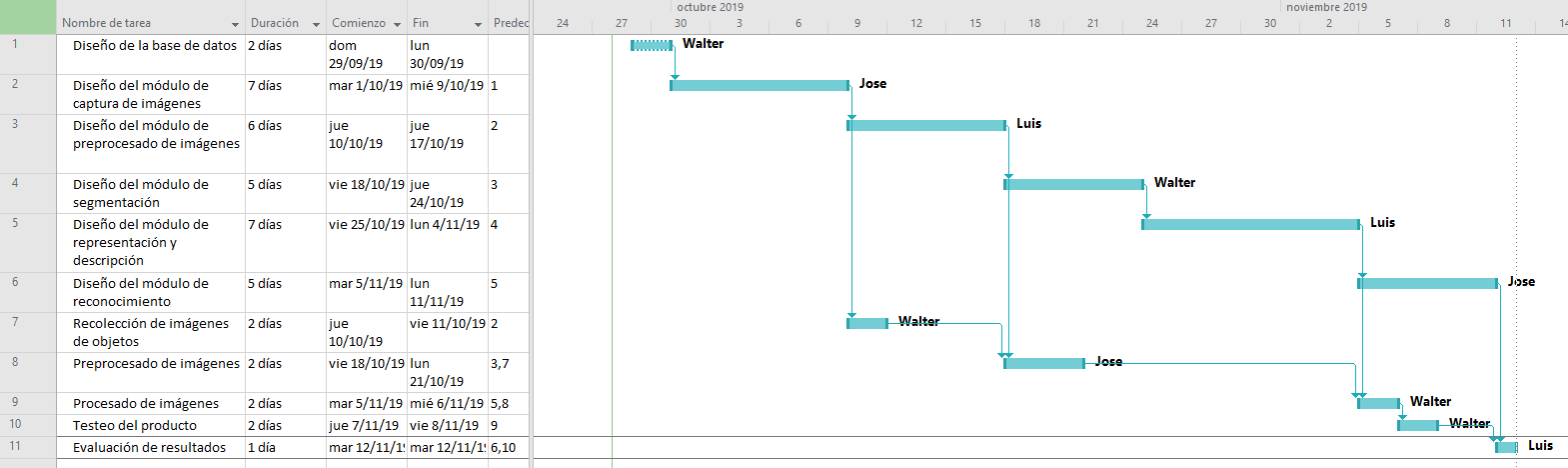


Una vez obtenemos un clasificador, entrenamos los siguientes teniendo en cuenta los casos negativos no descartados por los clasificadores anteriores.

**Ventajas**

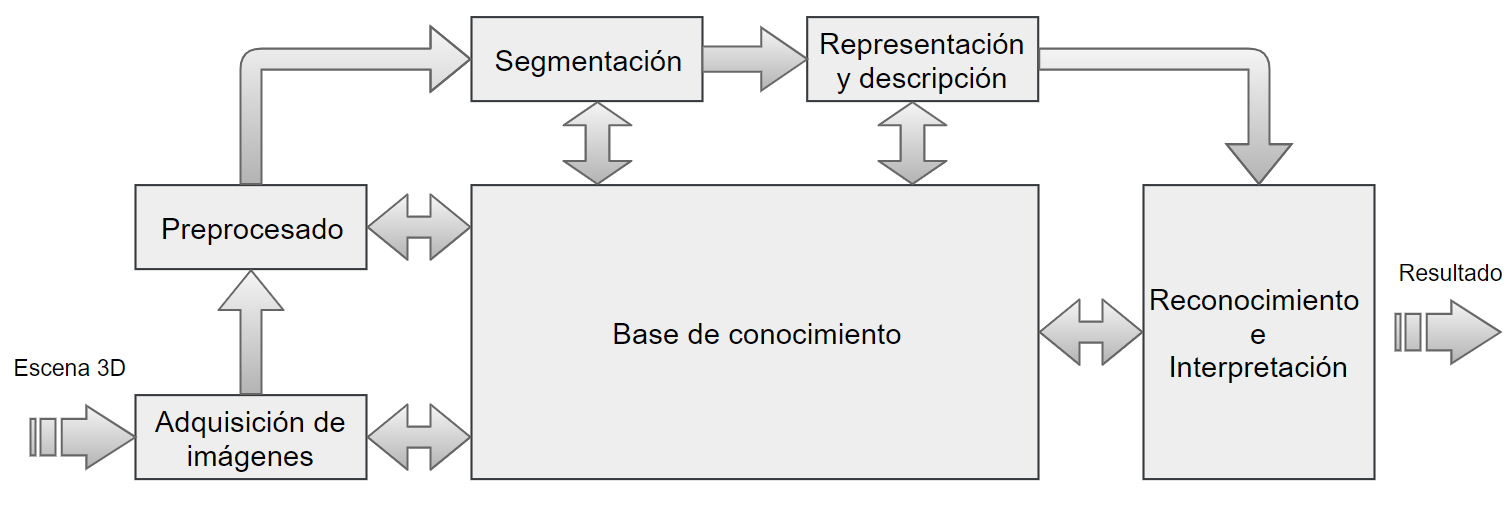
* Soporte Imágenes de alta calidad
* Para almacenar fotografías con una buena calidad se utilizan los formatos .bmp y .tiff
* Permite diferenciar mejor los aspectos tanto al color y las curvas de una imagen.
* Compatibilidad con librerias como OpenCV o OpenGl.

1. Plan para el desarrollo del sistema



1. descripción de la arquitectura del sistema

El proceso para el reconocimiento automático de objetivos se inicia con la Adquisición de la imagen de una escena en tres dimensiones, a continuación esta imagen se procesa para mejorar la calidad y eliminar posibles imperfecciones; el siguiente paso es separar el objeto de interés del fondo de la imagen, seguida de la extracción de sus características que describen al objeto (color, textura y geometría), para finalmente, comparar estas características con las de otros objetos que se tienen en la base de conocimiento y así determinar el tipo de objeto.

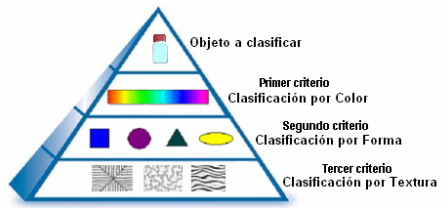


En cada una de las etapas anteriores se requiere de un conocimiento previo, como puede ser: el tamaño del objeto, la distancia a la que se encuentra, las condiciones ambientales, la inclinación, el número de objetos a reconocer en la imagen, entre otros. Toda esta información es necesaria para la utilización de determinadas técnicas y el desarrollo de algoritmos adicionales para el reconocimiento del objeto.

Para la implementación se utilizó una videocámara en la captación de la imagen y un láser para el cálculo de la distancia del objeto, a continuación se procesa esta imagen por medio de nuestro sistema de reconocimiento para su clasificación y finalmente se muestra en forma audible el nombre del objeto identificado y su distancia con respecto de la videocámara.



El método de reconocimiento se implementa en forma piramidal (figura 1.5.2) teniendo como primer criterio el color del objeto, como segundo criterio la forma del objeto y por último el reconocimiento por su textura. Con esta técnica se pretende mejorar el porcentaje de certeza para el reconocimiento del objeto utilizando tres técnicas de reconocimiento. Si al implementar el reconocimiento por color tenemos una certeza mayor al 90%, no será necesario utilizar las otras dos técnicas; de lo contrario, se utiliza el reconocimiento por forma y si no fueron suficientes los primeros dos métodos para reconocer con certeza el objeto en cuestión, se utiliza el reconocimiento por su textura.



El reconocimiento se realiza en un ambiente controlado como el salón de clases y con una buena iluminación ya que, como se verá en el capitulo 3, la iluminación juega un papel determinante en el éxito o fracaso del proyecto. Para ello fue necesario agregar luz incandescente (un foco de 100 wats) y apagar la luz fluorescente (luz blanca), cerrando la puerta y cubriendo las ventanas para que la luz externa no influya sobre la captación de la imagen.

Para evitar sombras y reflejos de luz, se utilizó un papel color negro. Además, el objeto a reconocer debe de estar en una posición en la cual se le puedan extraer de la mejor manera sus características y no debe de haber otros objetos transpuestos.



Descripción de procesos

**ADQUISICIÓN DE IMÁGENES**

Este proceso es de gran importancia y determinante para el éxito o fracaso de nuestro sistema de reconocimiento. Se dice que la captación de la imagen es el 50% del sistema y sería un error muy serio pensar que se puede compensar una imagen inadecuada con algún algoritmo; es mejor tener una buena calidad de imagen y utilizar menos algoritmos para su mejoramiento. El ojo humano tiene casi el mismo funcionamiento que una cámara de vídeo pero no se compara en cuanto a su definición de imágenes y al tiempo de procesamiento, ya que el ojo humano es uno de los órganos más especializados en el reconocimiento. Por todo esto, en este capítulo se menciona la anatomía y el funcionamiento del ojo humano, así como el funcionamiento de la cámara de vídeo, aspectos de iluminación y algunos otros conceptos importantes.

**Pre procesado**

La aparición de variaciones en intensidad debidas al ruido, por deficiencias en la iluminación o la obtención de imágenes de bajo contraste hace necesario un pre procesamiento de la imagen con el objetivo fundamental de corregir estos problemas, además de aplicar aquellas transformaciones a la imagen que acentúen las características que se deseen extraer de las mismas, de manera que se facilite las operaciones de las etapas posteriores.

**Segmentación**

En esta etapa se particiona o divide la imagen en sus componentes (objetos de interés) para su uso posterior. La segmentación autónoma es una de las tareas más difíciles del procesamiento de imágenes ya que esta etapa determina el éxito o el fracaso del proyecto. La segmentación puede ser orientada en regiones (área de la imagen en la que sus píxeles poseen propiedades similares de intensidad o de color) o a bordes (líneas que separan dos regiones). Tanto la detección de regiones como la de bordes implican una manipulación de la imagen original, donde los valores de los píxeles originales son modificados mediante ciertas operaciones de transformación u operadores.

**Representación y descripción**

En esta etapa, a la imagen le son extraídos un conjunto de características que describen las propiedades físicas de los objetos como son: el color, textura, área, perímetro, ancho, alto, promedio de intensidad, centroide, orientación, color, número de hoyos, entre otros.

**Reconocimiento e interpretación**

El último paso para el reconocimiento automático del objeto es la interpretación de los datos (descriptores) representados por características, para ello en el presente proyecto se implementó el clasificador “K vecinos más próximos”.

1. descripción de los datos del sistema

**Captación de la imagen por la cámara**

La captación de la imagen se realiza por medio un video, en donde mediante un script de Python obtenemos todos los fotogramas posibles, estos fotogramas serán filtrados, de modo que toda toma del objeto este dentro del a imagen.

**Iluminación**

La iluminación es el aspecto más decisivo de cualquier aplicación de visión artificial, si el objeto tiene una buena iluminación se pueden lograr muy buenos resultados en nuestro sistema de reconocimiento de objetos. Muchas aplicaciones buenas han fallado por la falta de una iluminación apropiada. En un sistema de visión artificial, la mejor imagen es aquella que tiene mayor contraste, donde las áreas de interés se destacan del fondo (background) que no tiene mayor importancia.

**Formatos de imágenes**

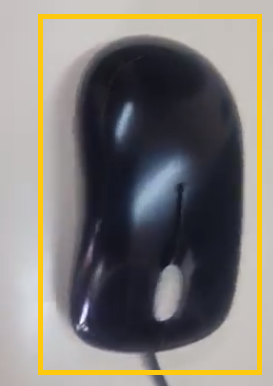
BMP. - El formato BMP (Bit Map) es el formato de las imágenes en bitmap de Windows. Aunque muy extendido, tiene la dificultad de la escasa compresión que realiza en los archivos por lo que ocupan mucho espacio. Pero el formato de Mapa de Bits tiene una importante característica a su favor, es que casi todos los usuarios tienen una PC con ambiente gráfico Windows y pueden leerlo desde un accesorio llamado “Paint”.

1. descripción del prototipo del sistema

El prototipo, será puesto a prueba con algunos objetos de sistema para este caso se eligió los monitores y los mouses del salón de clases, esto se hará mediante la grabación de un video, en donde se enfocará a los objetos mencionados y el sistema tendrá que identificarlos.



Demostración el prototipo, probado con un monitor.



Demostración el prototipo, probado con un mouse.

1. Bibliografía:

* <http://www.geintra-uah.org/system/files/private/Proyecto.pdf>

Este proyecto aborda el problema de la obtención de la posición y orientación métrica de los objetos tridimensionales en un entorno. Se propone un sistema de cámaras calibradas e interconectadas que forman un “Espacio Inteligente”.

* <https://upcommons.upc.edu/bitstream/handle/2099.1/9782/PFC_RogerGimeno.pdf>

Estudio de técnicas de reconocimiento facial

* <http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/mendieta_d_d/capitulo2.pdf>

Se menciona los conceptos y la teoría que fundamenta los procesos por los que pasa la imagen desde su ingreso hasta su reconocimiento.