**Visión por Computador**

**Diego Ramírez Gómez**

**Ángel Augusto Agudelo Zapata**

**Comunicaciones I**

**Universidad Tecnológica de Pereira**

**Ingeniera de Sistemas y Computación**

**Risaralda, Pereira**

**2016**

Introducción

La visión artificial o visión por computador es una disciplina científica que incluye métodos para adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información numérica o simbólica para que puedan ser tratados por un computador. Tal y como los humanos usamos nuestros ojos y cerebros para comprender el mundo que nos rodea, la visión por computador trata de producir el mismo efecto para que las computadoras puedan percibir y comprender una imagen o secuencia de imágenes y actuar según convenga en una determinada situación. Esta comprensión se consigue gracias a distintos campos como la geometría, la estadística, la física y otras disciplinas. La adquisición de los datos se consigue por varios medios como secuencias de imágenes, vistas desde varias cámaras de vídeo o datos multidimensionales desde un escáner médico.

El termino Visión por Computador ha sido muy utilizado en los últimos años y tiende a ser confundido con otros conceptos. Con el fin de aclarar este y otros términos se presenta en esta sección algunas definiciones.

Procesamiento Digital de Imágenes: Proceso mediante el cual se toma una imagen y se produce una versión modificada de esta imagen.

Análisis de Imágenes: Proceso mediante el cual a partir de una imagen se obtiene una medición, interpretación o decisión.

El análisis consiste en cinco etapas:

• Adquisición de la imagen: se obtiene la imagen adecuada del objeto de estudio. Dependiendo de la aplicación la imagen puede ser una fotografía, radiografía, termografía, etc.

• Preprocesamiento: con el fin de mejorar la calidad de la imagen obtenida se emplean ciertos filtros digitales que eliminan el ruido en la imagen o bien aumentan el contraste.

• Segmentación: Se identifica el objeto de estudio en la imagen.

•Medición (extracción de características): se realiza una medición objetiva de ciertos atributos de interés del objeto de estudio.

• Interpretación (clasificación): de acuerdo a los valores obtenidos en las mediciones se lleva a cabo una interpretación del objeto.

Reconocimiento de Patrones: Asignación de objetos (patrones) a diferentes clases a partir de mediciones de los objetos.

Hay muchas tecnologías que utilizan la visión por computador, entre las cuáles están: reconocimiento de objetos, detección de eventos, reconstrucción de una escena (mapping) y restauración de imágenes.

**Restauración de Imágenes**

Para la restauración de imágenes se utiliza el procesamiento digital de imágenes, que  es el conjunto de técnicas que se aplican a las imágenes digitales con el objetivo de mejorar la calidad o facilitar la búsqueda de información

**Filtros**

Es el conjunto de técnicas englobadas dentro del preprocesamiento de imágenes cuyo objetivo fundamental es obtener, a partir de una imagen origen, otra final cuyo resultado sea más adecuado para una aplicación específica mejorando ciertas características de la misma que posibilite efectuar operaciones del procesado sobre ella.

Los principales objetivos que se persiguen con la aplicación de filtros son:

* Suavizar la imagen: reducir la cantidad de variaciones de intensidad entre píxeles vecinos.
* Eliminar ruido: eliminar aquellos píxeles cuyo nivel de intensidad es muy diferente al de sus vecinos y cuyo origen puede estar tanto en el proceso de adquisición de la imagen como en el de transmisión.
* Realzar bordes: destacar los bordes que se localizan en una imagen.
* Detectar bordes: detectar los píxeles donde se produce un cambio brusco en la función intensidad.

Por tanto, se consideran los filtros como operaciones que se aplican a los píxeles de una imagen digital para optimizarla, enfatizar cierta información o conseguir un efecto especial en ella.

El proceso de filtrado puede llevarse a cabo sobre los dominios de frecuencia y/o espacio.

**Algunos Filtros Implementados**

**Blanco y Negro-** Para aplicar el filtro blanco y negro, se sacó el promedio de los valores de cada pixel y luego se compara con un valor determinado, si el promedio es menor al valor cambiamos el pixel a negro y si el promedio es mayor al valor cambiamos a blanco.

**Escala de Grises-**Para el filtro de escala de grises lo que se necesita es simplemente obtener el promedio de los valores de r, g y b de cada pixel y luego modificar ese pixel colocando el valor del promedio en los valores r, g y b.

**Umbrales-**El filtro de umbrales es una combinación de los dos anteriores, solo que aquí aplicamos un valor máximo y uno mínimo para determinar cuándo convertir a blanco o negro el pixel

**Negativo-**EL filtro de negativo es muy simple, se resta el valor r, g y b de 255, y luego colocamos ese nuevo valor en el pixel.

**Suavizado-** Se logra suavizar la imagen obteniendo cada pixel, así mismo como su pixel en dirección norte, sur, este y oeste, y sacando el promedio de estos

.

**Sepia-**El filtro de sepia implica alterar los valores r, g y b. El valor de r se eleva, el valor de g se eleva menos drásticamente, y en el b se disminuye.

**Reconocimiento de objetos –** en visión por computadoras es la tarea para encontrar e identificar objetos en una imagen o secuencia de vídeo. Los humanos reconocemos una multitud de objetos en imágenes con poco esfuerzo, a pesar del hecho que la imagen del objeto puede variar un poco en diferentes puntos de vista, en diferentes tamaños o escalas e incluso cuando están trasladados o rotados. Los objetos pueden ser reconocidos cuando están parcialmente obstruidos desde una vista. No obstante esta tarea es un desafío para los sistemas de visión por computadoras. Para este problema se han implementado muchos métodos durante múltiples décadas entre los cuales se encuentra la detección de bordes.

**Detección de Bordes**

La detección de bordes es una herramienta fundamental en el [procesamiento de imágenes](https://es.wikipedia.org/wiki/Procesamiento_de_im%C3%A1genes) y en [visión por computadora](https://es.wikipedia.org/wiki/Visi%C3%B3n_por_computadora), particularmente en las áreas de [detección](https://es.wikipedia.org/w/index.php?title=Detecci%C3%B3n_de_caracter%C3%ADsticas&action=edit&redlink=1) y [extracción de características](https://es.wikipedia.org/w/index.php?title=Extracci%C3%B3n_de_caracter%C3%ADsticas&action=edit&redlink=1), que tiene como objetivo la identificación de puntos en una [imagen digital](https://es.wikipedia.org/wiki/Imagen_digital) en la que el [brillo de la imagen](https://es.wikipedia.org/wiki/Luminosidad) cambia drásticamente o, más formalmente, tiene discontinuidades.

Hay muchos métodos para la detección de bordes, pero sobre todo difieren en los tipos de filtros de suavizado que se aplican y la forma en que se calculan las medidas de resistencia de los bordes.

**Detector de bordes de Canny**

El detector de bordes de Canny es un operador de detección de bordes que usa un algoritmo multi-etapa. Fue desarrollado por Jhon F. Canny en 1986.

La detección de bordes de Canny es una técnica para extraer información estructural útil de diferentes objetos de visión y reducir drásticamente la cantidad de datos a procesar. Ha sido ampliamente aplicada en los diferentes sistemas de visión por ordenador. Canny encontró que los requerimientos para la detección de bordes en diferentes sistemas de visión son relativamente similares, así que el criterio general para la detección de bordes incluye:

1. **Buena detección:** el algoritmo debe marcar el mayor número real de bordes en la imagen como sea posible.
2. **Buena localización:** los bordes de marca deben estar lo más cerca posible del borde de la imagen real.
3. **Respuesta mínima:** El borde de una imagen sólo debe ser marcado una vez, y siempre que sea posible, el ruido de la imagen no debe crear falsos bordes.
   1. Algoritmo

El proceso de detección de bordes de Canny puede descomponerse en 5 pasos diferentes:

1. Aplicar ***Gaussian filter*** (filtro gaussiano) para suavizar la imagen con el fin de eliminar el ruido.
2. Encontrar los ***intensity gradients*** (gradientes de intensidad) de la imagen.
3. Aplicar ***non-maximum suppression*** (supresión no máxima) para deshacerse de la respuesta parásita en la detección de bordes.
4. Aplicar ***double threshold*** (doble umbral) para determinar posibles bordes.
5. Finalizar la detección de bordes mediante la supresión de todos los otros bordes que son débiles y no conectan a los bordes fuertes.
   1. Gaussian filter

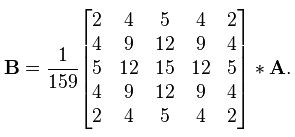
Puesto que todos los resultados de la detección de bordes son fácilmente afectados por el ruido de la imagen, es esencial filtrar el ruido para evitar las falsas detecciones. Para suavizar la imagen, se aplica un filtro gaussiano de convolución con la imagen. Este paso suaviza ligeramente la imagen para reducir los efectos del ruido en el detector de bordes.

Es importante entender que la selección del tamaño del kernel gaussiano afectará el rendimiento del detector. Cuanto mayor sea el tamaño, menor es la sensibilidad del detector al ruido. Además, el error de localización para detectar el borde aumentará ligeramente con el aumento del tamaño del núcleo de filtro de Gauss. 5 × 5 es un buen tamaño para la mayoría de los casos, pero esto también variará dependiendo de situaciones específicas.

Ecuación para el kernel del filtro gaussiano de tamaño (2k+1) X (2k+1).

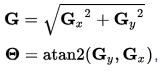


Kernel empleado en el algoritmo.

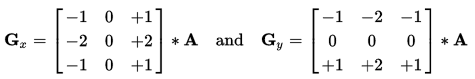


* 1. Intensity gradients

Un borde de una imagen puede apuntar en una variedad de direcciones, por lo que el algoritmo de Canny utiliza cuatro filtros para detectar bordes horizontales, verticales y diagonales en la imagen borrosa. El operador de detección de borde (***Sobel***) devuelve un valor de la primera derivada en la dirección horizontal (Gx) y la dirección vertical (Gy). A partir de este gradiente y la dirección del borde se pueden determinar:



***El operador de Sobel*** utiliza dos núcleos de 3 × 3 que se convolucionan con la imagen original para calcular aproximaciones de las derivadas ,una para los cambios horizontales, y una para los verticales. Si definimos la imagen de origen como A, y Gx y Gy dos imágenes que en cada punto contienen las aproximaciones de la derivadas horizontales y verticales respectivamente, los cálculos son los siguientes:



donde \* denota la operación de convolución.

* 1. Non-maximum suppression

La supresión no máxima es una técnica de adelgazamiento de borde.

la supresión no máxima es aplicada para "adelgazar" del borde. Después de aplicar el cálculo del gradiente, el borde extraído del valor de gradiente es todavía muy borroso. En lo que respecta al criterio 3, sólo debe haber una respuesta precisa a la orilla. Por lo tanto la supresión no máxima puede ayudar a suprimir todos los valores de gradiente a 0, excepto el máximo local, lo que indica la ubicación con el cambio más agudo del valor de intensidad. El algoritmo para cada píxel de la imagen en el gradiente es:

1. Comparación de la resistencia del borde del píxel actual con la resistencia del borde del píxel en las direcciones de gradiente positivos y negativos.
2. Si la intensidad de borde del píxel actual es el más grande en comparación con los otros píxeles de la máscara con la misma dirección (es decir, el píxel que está apuntando en la dirección Y, se compara con el píxel por encima y por debajo en el eje vertical), se conservará el valor. De lo contrario, se suprimirá el valor.

En algunas implementaciones, el algoritmo clasifica las direcciones gradiente continuas en un pequeño conjunto de direcciones discretas, y entonces mueve un filtro de 3x3 sobre la salida de la etapa anterior (es decir, la resistencia del borde y las direcciones de gradiente). En cada píxel, se suprime la resistencia del borde del píxel central (estableciendo su valor a 0) si su magnitud no es mayor que la magnitud de los dos vecinos en la dirección del gradiente. Por ejemplo:

* si el ángulo de pendiente redondeado es de 0 ° (es decir, el borde está en la dirección norte-sur), se considera que el punto está en el borde si la magnitud de su gradiente es mayor que las magnitudes de los píxeles en las direcciones este-oeste.
* si el ángulo de pendiente redondeado es de 90 ° (es decir, el borde está en la dirección este-oeste), se considera que el punto está en el borde si la magnitud de su gradiente es mayor que las magnitudes de los píxeles en las direcciones norte-sur.
* si el ángulo de pendiente redondeado es de 135 ° (es decir, el borde está en la dirección noreste-suroeste), se considera que el punto está en el borde si la magnitud de su gradiente es mayor que las magnitudes de los píxeles en las direcciones noroeste-sureste.
* si el ángulo de pendiente redondeado es de 45 ° (es decir, el borde está en la dirección noroeste-sureste), se considera que el punto está en el borde si la magnitud de su gradiente es mayor que las magnitudes de los píxeles en las direcciones noreste-suroeste.
  1. Double threshold

Después de la aplicación de la supresión no máxima, los píxeles del borde son bastante exactos para presentar el borde real. Sin embargo, en este punto todavía hay algunos píxeles del borde causados por el ruido y la variación de color. Con el fin de deshacerse de las respuestas parásitas de estos factores, es esencial filtrar los píxeles del borde con valor de gradiente débil y preservar los de alto valor. Así, dos valores de umbral se fijan para aclarar los diferentes tipos de píxeles del borde, uno se llama alto valor de umbral y el otro se llama bajo valor de umbral. Si el valor de gradiente del píxel de borde es mayor que el valor de umbral alto, dicho píxel se marca como fuerte. Si el valor de gradiente del píxel de borde es menor que el valor de umbral alto y más grande que el valor de umbral bajo, se marca como débil. Si el valor de píxel es menor que el valor de umbral bajo, será suprimido. Los dos valores de umbral son empíricamente determinados, y necesitan ser definidos cuando se aplican a diferentes imágenes.

**Reconocimiento Facial**

El reconocimiento facial es una aplicación dirigida por [ordenador](https://es.wikipedia.org/wiki/Ordenador) que identifica automáticamente a una persona en una [imagen digital](https://es.wikipedia.org/wiki/Imagen_digital). Esto es posible mediante un análisis de las características faciales del sujeto extraídas de la imagen o de un fotograma clave de una fuente de video, y comparándolas con una base de datos. Se ha convertido en los últimos años en un área de investigación activa que abarca diversas disciplinas, como [procesado de imágenes](https://es.wikipedia.org/wiki/Procesado_de_imagen), [reconocimiento de patrones](https://es.wikipedia.org/wiki/Reconocimiento_de_patrones), [visión por ordenador](https://es.wikipedia.org/wiki/Visi%C3%B3n_por_computadora) y [redes neuronales](https://es.wikipedia.org/wiki/Red_neuronal_artificial). Involucra tanto a investigadores del área de informática como a neurocientíficos y psicólogos. Se podría considerar también dentro del campo de [reconocimiento de objetos](https://es.wikipedia.org/wiki/Reconocimiento_de_objetos), donde la cara es un objeto tridimensional sujeto a variaciones de iluminación, pose, etc., y ha de ser identificada basada en su proyección 2D (excepto cuando se utilizan técnicas 3D).

El objetivo de un sistema de reconocimiento facial es, generalmente, el siguiente: dada una imagen de una cara "desconocida", o imagen de test, encontrar una imagen de la misma cara en un conjunto de imágenes "conocidas", o imágenes de entrenamiento. La gran dificultad añadida es la de conseguir que este proceso se pueda realizar en tiempo real. El sistema identificará las caras presentes en imágenes o videos automáticamente. Puede operar en dos modos:

* Verificación o autentificación de caras: compara una imagen de la cara con otra imagen con la cara de la que queremos saber la identidad. El sistema confirmará o rechazará la identidad de la cara.
* Identificación o reconocimiento de caras: compara la imagen de una cara desconocida con todas las imágenes de caras conocidas que se encuentran en la base de datos para determinar su identidad.

Por su naturaleza amigable, este tipo de sistemas siguen siendo atractivos a pesar de la existencia de otros métodos muy fiables de identificación personal [biométricos](https://es.wikipedia.org/wiki/Biometr%C3%ADa), como el [análisis de huellas dactilares](https://es.wikipedia.org/wiki/Huella_dactilar) y el [reconocimiento del iris](https://es.wikipedia.org/wiki/Reconocimiento_de_iris).

Algoritmo

El proceso consta de cuatro módulos principales:

1. [**Detección de la cara**](https://es.wikipedia.org/wiki/Detecci%C3%B3n_de_caras): detecta que hay una cara en la imagen, sin identificarla. Si se trata de un video, también podemos hacer un seguimiento de la cara. Proporciona la localización y la escala a la que encontramos la cara.
2. **Alineación de la cara**: localiza las componentes de la cara y, mediante transformaciones geométricas, la normaliza respecto propiedades geométricas, como el tamaño y la pose, y fotométricas, como la iluminación. Para normalizar las imágenes de caras, se pueden seguir diferentes reglas, como la distancia entre las pupilas, la posición de la nariz, o la distancia entre las comisuras de los labios. También se debe definir el tamaño de las imágenes y la gama de colores. Normalmente, para disminuir la carga computacional del sistema, se acostumbra a utilizar imágenes pequeñas en escala de grises. A veces también se realiza una ecualización del histograma.
3. [**Extracción de características**](https://es.wikipedia.org/w/index.php?title=Extracci%C3%B3n_de_caracter%C3%ADsticas&action=edit&redlink=1): proporciona información para distinguir entre las caras de diferentes personas según variaciones geométricas o fotométricas.
4. **Reconocimiento**: el vector de características extraído se compara con los vectores de características extraídos de las caras de la base de datos. Si encuentra uno con un porcentaje elevado de similitud, nos devuelve la identidad de la cara; si no, nos indica que es una cara desconocida.

Opencv nos facilita la terea de detectar rostros pues ya cuenta con clasificadores entrenados para esta tarea almacenados en archivos XML, pero en caso de que lo necesitemos podemos crear nuestros propios clasificadores, además de contar con todas las funciones necesarias para esta tarea.

**Detección de rostros en opencv**

Para detectar un rostro primero debemos procesar la imagen en la cual deseamos buscar un rostro, luego de cargar la imagen debemos aplicar los siguientes pasos:

Convertir la imagen a escala de grises, necesario para el correcto funcionamiento de los algoritmos de detección de rostros usados por opencv. Para convertir una imagen a escala de grises u otro formato contamos con la función cvtColor.

Lo siguiente que debemos hacer es aplicar ecualización de histograma a la imagen en grises para estandarizar el contraste y brillo de la imagen, esto para que distintas condiciones de iluminación no afecten la detección del rosto en la imagen. Con la imagen procesada ahora debemos cargar el detector que deseamos utilizar,

OpencV nos proporciona una variead de cascadas para detección tanto de rostros, asi como de ojos, nariz y boca, Incluso de cuerpos enteros

**Detección de rostros en tiempo real**

Para detectar rostros en tiempo real solo debemos leer las imágenes de la webcam y aplicarle a cada una de ellas el procedimiento descrito anteriormente, para ver cómo usar la webcam.