

Cuestionario 1

Luis Antonio Ortega Andrés

27 de octubre de 2019

1. Pregunta 1

Diga en una sola frase cuál cree que es el objetivo principal de la Visión por Computador.

El objetivo principal de la visión por computador es el uso de técnicas matemáticas y algorítmicas para el tratamiento de imágenes y la obtención de información sobre estas.

Diga también cuál es la principal propiedad de las imágenes de cara a la creación algoritmos que la procesen.

La principal propiedad de las imágenes es que parte de la información de un píxel se encuentra en los píxeles cercanos a este. Esto también influye a la hora de tratar con ruido en la imagen.

2. Pregunta 2

Expresar las diferencias y semejanzas entre las operaciones de correlación y convolución. Dar una interpretación de cada una de ellas que en el contexto de uso en visión por computador.

Ambas operaciones son transformaciones que utilizan la información local de la imagen. Es decir, actualizan el valor de un píxel utilizando el de los píxeles cercanos. Ambas utilizan una matriz cuadrada de orden impar de números reales llamada *máscara* que define la operación.

Las operaciones que las definen son similares. Dada una máscara K , y una imagen M , denotamos \star a la operación de convolución y \otimes a la de correlación.

$$H \star M[u, v] = \sum_i \sum_j H[u, v] M[u - i, v - j]$$

$$H \otimes M[u, v] = \sum_i \sum_j H[u, v] M[u + i, v + j]$$

Ambas operaciones coinciden cuando la máscara es simétrica y en otro caso, invertir las filas y las columnas de la máscara de una, resulta la máscara de la otra.

En el contexto de visión por computador, la convolución se utiliza para aplicar filtros a la imagen debido a que tiene mejores propiedades que la correlación, algunas de estas son ser asociativa, conmutativa y bilineal. La correlación se utiliza como un operador de búsqueda de patrones dentro de una imagen, esto se debe a que al hacer la correlación de una imagen con un patrón obtendremos respuestas altas en aquellas zonas de la imagen que sean similares a dicho patrón.

3. Pregunta 3

¿Cuál es la diferencia “esencial” entre el filtro de convolución y el de mediana? Justificar la respuesta.

La diferencia principal entre ambos filtros es que la mediana no es una función lineal sobre las imágenes. Para justificar la respuesta, veamos la demostración de que la convolución si lo es y un contraejemplo para la mediana.

Tomamos una máscara o kernel K , dos imágenes A y B y una constante a .

$$\begin{aligned} K \star (aA + B) &= \sum_{u,v} K[u,v](aA + B)[i-u, j-v] = \sum_{u,v} aK[u,v]A[i-u, j-v] + \sum_{u,v} K[u,v]B[i-u, j-v] \\ &= a \sum_{u,v} K[u,v]A[i-u, j-v] + \sum_{u,v} K[u,v]B[i-u, j-v] = a(K \star A) + K \star B \end{aligned}$$

Con esto vemos que la convolución es lineal. Veamos ahora un contraejemplo de la linealidad de la mediana. Sean $A = [1, 2, 3]$, $B = [3, 2, 1]$ y K una máscara mediana de tamaño 3, con bordes a 0.

$$K(A) + K(B) = [1, 2, 2] + [2, 2, 1] = [3, 4, 3]$$

$$K(A + B) = K([4, 4, 4]) = [4, 4, 4]$$

4. Pregunta 4

Identifique el “mecanismo concreto” que usa un filtro de máscara para transformar una imagen.

El mecanismo concreto que utiliza un filtro de máscara para transformar una imagen es la información local que tiene esta. El valor de un píxel tras aplicar la máscara solo depende de los valores de su entorno en la imagen original. La cantidad de vecinos que intervienen en el cálculo viene dado por el tamaño de la máscara.

5. Pregunta 5

¿De qué depende que una máscara de convolución pueda ser implementada por convoluciones 1D? Justificar la respuesta.

Para que una máscara A de convolución 2D de dimensiones $n \times m$, se pueda implementar mediante convoluciones 1D, esta se debe poder descomponer en dos vectores de la forma $A = a_1 a_2^T$, con $a_1 \in \mathbb{R}^n$ y $a_2 \in \mathbb{R}^m$.

$$(A \star B)[u, v] = \sum_i \sum_j A[i, j] B[u-i, v-j] = \sum_i \sum_j a_1[i] a_2^T[j] B[u-i, v-j] = \sum_i a_1[i] \sum_j a_2^T[j] B[u-i, v-j]$$

Veamos entonces que la condición necesaria y suficiente para que esto ocurra es que el rango de la matriz A sea 1.

Supongamos que la matriz A es separable, es decir, $\exists u, v$ tal que $A = uv^T$. Veamos A como una aplicación $A : \mathbb{R}^m \rightarrow \mathbb{R}^n$. Consistente en multiplicar el vector por la derecha.

Entonces $A(x) = uv^T x = \langle x, v \rangle u$, donde $\langle x, v \rangle \in \mathbb{R} \implies \dim \text{Im}(A) = 1 \implies \text{rango}(A) = 1$.

Asumamos ahora que el rango de la matriz A es 1, entonces $\exists v \in \mathbb{R}^n$ tal que $\forall u \in \mathbb{R}^m \exists k \in \mathbb{R}$ que cumple $A(u) = Au = kv$. En particular esto ocurre para los vectores de la base usual. Esto implica que las columnas de A son proporcionales a $v \implies \exists w$ tal que $A = vw^T$

Para calcular dichos vectores dada una máscara 2D, seguimos los siguientes pasos:

1. Comprobamos que el rango de la matriz es 1.
2. Buscamos una fila u y una columna w no nulas (deben existir por tener rango 1, y el resto de filas y columnas serán proporcionales a estas).
3. Realizamos el producto de u y w resultando una matriz proporcional a la original.
4. Dividimos uno de los dos vectores entre dicha constante de proporcionalidad.

6. Pregunta 6

Identificar las diferencias y consecuencias desde el punto de vista teórico y de la implementación entre:

- a) Primero alisar la imagen y después calcular las derivadas sobre la imagen alisada
- b) Primero calcular las imágenes derivadas y después alisar dichas imágenes.

Justificar los argumentos.

En el plano teórico de la convolución, el resultado es el mismo en ambas operaciones, debido a que la operación es conmutativa.

$$X \star Y = \sum_{u,v} X[u,v]Y[a-u,b-v] = \sum_{u,v} X[a-u,b-v]Y[u,v] = Y \star X$$

Entonces, sea G un filtro de alisado, D una máscara de derivada y A una imagen.

$$G \star (D \star A) = D \star (G \star A)$$

Sin embargo, en el caso práctico, ambas soluciones pueden no dar el mismo resultado debido a errores de redondeo o el tipo de borde utilizado.

Cuantas más operaciones se realicen, más estaremos sujetos a este tipo de fallos, por lo tanto, es preferible realizar el alisado antes de aproximar las derivadas, ya que si primero calculamos las derivadas, luego tendremos que alisar cada una de ellas.

7. Pregunta 7

Identifique las funciones de las que podemos extraer pesos correctos para implementar de forma eficiente la primera derivada de una imagen. Suponer alisamiento Gaussiano.

En el caso de estar calculando la primera derivada con respecto de x , la función de la que podemos extraer pesos correctos para implementarlo de forma eficiente es:

$$F(x,y) = f_1(x)f_2(y) \text{ con } f_1(x) = \frac{-x}{\sigma_x^2}g_{\sigma_x}(x), f_2(y) = g_{\sigma_y}(y)$$

Donde

$$g_{\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-x^2}{2\sigma^2}}$$

Siendo análogo para la derivada respecto de y .

Para ver esto, utilizamos la respuesta del apartado anterior. Queremos realizar primero el alisamiento y luego realizar la derivada. Además, buscamos que el filtro resultante sea separable, de forma que se pueda convolucionar por filas y por columnas. Sea G el filtro Gaussiano y M nuestra imagen.

$$\frac{\partial}{\partial x}(G \star M) = \frac{\partial G}{\partial x} \star M$$

Por esto, consideramos las derivadas parciales de G .

Definimos la función Gaussiana de 2 variables, con media 0 y desviaciones típicas σ_x, σ_y .

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)} = \frac{1}{\sqrt{2\pi}\sigma_x} e^{-\frac{x^2}{2\sigma_x^2}} + \frac{1}{\sqrt{2\pi}\sigma_y} e^{-\frac{y^2}{2\sigma_y^2}} = g_{\sigma_x}(x)g_{\sigma_y}(y)$$

Como vemos, es una función separable, de forma que sus parciales también lo serán. Las calculamos.

$$\frac{\partial g_{\sigma}}{\partial x}(x) = \frac{-2x}{2\sigma^2} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} = \frac{-x}{\sigma^2} g_{\sigma}(x)$$

Si derivamos en la expresión de la Gaussiana de 2 variables.

$$\begin{aligned} \frac{\partial g}{\partial x}(x, y) &= g'_{\sigma_x}(x)g_{\sigma_y}(y) = \frac{-x}{\sigma^2} g_{\sigma_x}(x)g_{\sigma_y}(y) \\ \frac{\partial g}{\partial y}(x, y) &= g_{\sigma_x}(x)g'_{\sigma_y}(y) = g_{\sigma_x}(x)\frac{-y}{\sigma^2} g_{\sigma_y}(y) \end{aligned}$$

8. Pregunta 8

Identifique las funciones de las que podemos extraer pesos correctos para implementar de forma eficiente la Laplaciana de una imagen. Suponer alisamiento Gaussiano.

Dada una imagen M , consideramos su Laplaciano

$$\Delta(M) = \frac{\partial^2}{\partial x^2}M + \frac{\partial^2}{\partial y^2}M$$

Seguimos el mismo razonamiento que en la pregunta anterior, calculando ahora las segundas derivadas de $g(x, y)$.

$$\begin{aligned} \frac{\partial^2}{\partial x^2}g(x, y) &= \frac{\partial}{\partial x}\left(\frac{\partial}{\partial x}g(x, y)\right) = \frac{x^2 - \sigma_x^2}{\sigma_x^4}g(x, y) = \frac{x^2 - \sigma_x^2}{\sigma_x^4}g_{\sigma_x}(x)g_{\sigma_y}(y) \\ \frac{\partial^2}{\partial y^2}g(x, y) &= \frac{\partial}{\partial y}\left(\frac{\partial}{\partial y}g(x, y)\right) = \frac{y^2 - \sigma_y^2}{\sigma_y^4}g(x, y) = \frac{y^2 - \sigma_y^2}{\sigma_y^4}g_{\sigma_x}(x)g_{\sigma_y}(y) \end{aligned}$$

Por lo tanto, aunque el operador Laplaciano no sea separable, las segundas derivadas si lo son, asique podemos extraer los pesos de cada una de ellas de forma eficiente. Luego calculamos el Laplaciano sumando los resultados.

9. Pregunta 9

Suponga que le piden implementar de forma eficiente un algoritmo para el cálculo de la derivada de primer orden sobre una imagen usando alisamiento Gaussiano. Enumere y explique los pasos necesarios para llevarlo a cabo.

El filtro Gaussiano puede estar especificado de dos formas distintas: por los parámetros que lo determinan (σ_x, σ_y) o como una matriz directamente (ya sea separada en 2 máscaras 1D o una máscara 2D). En el caso de tener el filtro a partir de sus parámetros, utilizamos la expresión separable del filtro Gaussiano y muestreamos en puntos equi-espaciados. Los pasos serían:

1. Calcular el tamaño que tendrán las máscaras 1D. Para ello buscamos que gran parte de la densidad de la función Gaussiana quede dentro. Para esto tomamos el intervalo $[-3\sigma, 3\sigma]$. El cual contiene mas del 99 % de la densidad.
2. Calculamos los valores de las máscaras a partir de la expresión de la derivada del Ejercicio 7.
3. Normalizamos el filtro multiplicando por σ .
4. Aplicar las máscaras 1D con una convolución a la imagen.

En caso de tener el filtro Gaussiano en forma de matriz 2D o como 2 vectores 1D, debemos calcular un nuevo filtro. Para lograr esto se puede utilizar una máscara de Sobel. Los pasos serían:

1. Realizamos la convolución de los filtros Gaussianos y los de derivadas.
2. Normalizamos el filtro multiplicando por σ .
3. Convolucionamos los filtros por filas y columnas con la imagen.

10. Pregunta 10

Identifique semejanzas y diferencias entre la pirámide gaussiana y el espacio de escalas de una imagen, ¿cuándo usar una u otra? Justificar los argumentos.

Ambos métodos nos permiten identificar elementos dentro de una imagen, para ello construyen modelos multi-escala de la misma, aunque no lo utilizan de la misma forma.

La pirámide Gaussiana permite visualizar la imagen en distintos tamaños, para ello hace uso de filtros de alisado (perdiendo las frecuencias altas), permitiendo obtener varias representaciones de la misma imagen con distinto nivel de detalle. El espacio de escalas emplea filtros de Laplaciana-de-Gaussiana para construir distintas «escalas» (no las muestra) que utiliza para encontrar elementos en la imagen (en forma de máximos locales entre las escalas).

Si queremos detectar características o patrones en una imagen dependiendo del nivel de detalle, o reducir su tamaño evitando el «aliasing», debemos utilizar una pirámide Gaussiana.

En cambio, si queremos detectar elementos y regiones en la imagen, debemos utilizar un espacio de escalas junto con un algoritmo de supresión de no-máximos para evitar el ruido presente en la imagen.

11. Pregunta 11

¿Bajo qué condiciones podemos garantizar una perfecta reconstrucción de una imagen a partir de su pirámide Laplaciana? Dar argumentos y discutir las opciones que considere necesario.

De forma teórica, siempre se puede garantizar la reconstrucción de la imagen. En la práctica, el procedimiento está sujeto a errores de cálculo con flotantes. Para hacerlo, debemos aplicar el algoritmo inverso al de la construcción de la pirámide Laplaciana.

Dada la pirámide Laplaciana de una imagen $\{L_1, \dots, L_n\}$, donde L_n corresponde con el nivel más bajo de la pirámide Gaussiana $\{G_1, \dots, G_n\}$ utilizada para construir la Laplaciana, $L_n = G_n$. Además, $L_i = G_i - f(G_{i+1})$ donde f corresponde a la función que aumenta el tamaño de la imagen.

Despejando tenemos $G_i = L_i + f(G_{i+1})$, luego podemos reconstruir la imagen original G_1 .

12. Pregunta 12

¿Cuáles son las contribuciones más relevantes del algoritmo de Canny al cálculo de los contornos sobre una imagen? ¿Existe alguna conexión entre las máscaras de Sobel y el algoritmo de Canny? Justificar la respuesta.

Las contribuciones más relevantes del algoritmo de Canny son que cumple los siguientes criterios:

1. Buena detección de bordes con pocos errores.
2. Los puntos de borde detectados se encuentran posicionados muy cerca del centro del borde real.
3. Cada borde de la imagen solo es marcado una vez. El ruido presente en la imagen no debe crear falsos bordes.

Para cumplir estos criterios el algoritmo de Canny realiza los siguientes pasos:

1. Aplica un filtro Gaussiano para eliminar el ruido.
2. Calcula el gradiente de la imagen.
3. Aplica supresión de no-máximos para eliminar falsos positivos.
4. Define dos umbrales, con ellos construye los bordes empezando por los puntos con valor por encima del umbral alto y continua añadiendo puntos si son mayores al umbral bajo.

Debido a que las máscaras de Sobel calculan una aproximación del gradiente de la imagen en cada punto, en el algoritmo de Canny podemos obtener el gradiente utilizándolas. OpenCV realiza esta operación por defecto.

13. Pregunta 13

Identificar pros y contras de k-medias como mecanismo para crear un vocabulario visual a partir del cual poder caracterizar patrones. ¿Qué ganamos y qué perdemos? Justificar los argumentos.

El método k-medias para construir un vocabulario visual nos permite encontrar grupos de características similares, es decir, realizar un «clustering» de los elementos de la imagen.

Algunas ventajas de este método son:

- Es relativamente sencillo de implementar.
- Escala bien a conjuntos de datos grandes.
- Se adapta con facilidad a nuevos ejemplos.

Algunas de sus desventajas son:

- Requiere fijar el valor de «k» manualmente. Este parámetro se debe encontrar mediante experimentación y en caso de no ser apropiado, puede dar malos resultados.
- Intenta crear «clusters» de tamaño y forma similar, independientemente de como estén distribuidos los datos.
- Puede caer en óptimos locales.

14. Pregunta 14

Identifique pros y contras del modelo de “Bolsa de Palabras” como mecanismo para caracterizar el contenido de una imagen. ¿Qué ganamos y que perdemos? Justificar los argumentos.

El modelo «Bolsa de Palabras» nos permite representar la imagen como un único vector de «palabras». Para ello se vale de un «vocabulario» previamente definido, que corresponderá con una serie de características que queremos buscar en la imagen. Luego necesitaremos algún algoritmo que detecte dichas regiones en la imagen, por ejemplo, el algoritmo k-medias.

El modelo no solo tiene en cuenta que «palabras» de dicho vocabulario aparecen en la imagen, si no cuántas veces lo hacen.

Entre sus ventajas se encuentran:

- Agrupa características parecidas y separa las diferentes, esto reduce su sensibilidad al ruido y favorece la discriminación.
- Permite representar la imagen en un único vector de forma compacta y eficiente.
- La detección de regiones no depende de la escala o la orientación.

Desventajas de este modelo son las siguientes:

- Es necesario construir un vocabulario de palabras visuales.
- El modelo no tiene en cuenta la localización de las características. Se pierde la información espacial de las mismas, lo cual puede ser clave a la hora de analizar la imagen.
- Como utiliza el algoritmo k-medias, está sujeto a las mismas desventajas que este.

15. Pregunta 15

Suponga que dispone de un conjunto de imágenes de dos tipos de clases bien diferenciadas. Suponga que conoce como implementar de forma eficiente el cálculo de las derivadas hasta el orden N de la imagen. Describa como crear un algoritmo que permita diferenciar, con garantías, imágenes de ambas clases. Justificar cada uno de los pasos que proponga.

Para cada una de las imágenes de las que disponemos su clasificación, seguimos la siguiente estrategia:

1. Utilizar el algoritmo SIFT, para extraer elementos de la imagen. Aquí es donde aprovechamos el cálculo eficiente de las derivadas.
2. Utilizar el algoritmo «k-medias» para agrupar los elementos y construir el «vocabulario» visual.
3. Utilizamos el modelo de la «Bolsa de Palabras» y almacenamos el vector de «palabras» junto con la clase a la que pertenece la imagen.

Para cada imagen que queramos clasificar, realizamos los mismos pasos y a la hora de decidir a que clase pertenece, buscamos a su vecino más cercano.

16. Bibliografía

1. Canny Edge Detector - Wikipedia.
https://en.wikipedia.org/wiki/Canny_edge_detector
2. Bolsa de Palabras - Wikipedia.
https://en.wikipedia.org/wiki/Bag-of-words_model
3. K-Means - Google Developers.
<https://developers.google.com/machine-learning/clustering/algorithm/advantages-disadvantages>
4. SIFT - Wikipedia.
https://en.wikipedia.org/wiki/Scale-invariant_feature_transform