

Histograma de gradientes orientados

Por Esteban Acevedo y Jonathan Cardona.

1. ¿Qué es?

El histograma de gradientes orientados (HOG) es un descriptor de características que se utiliza en la visión por computadora y el procesamiento de imágenes con el fin de detectar objetos . Es una técnica que cuenta eventos de orientación de gradiente en una porción específica de una imagen o región de interés.



Figura 1.1 Imagen original antes de ser procesada.

2. Historia

Robert K. McConnell de Wayland Research Inc. describió por primera vez los conceptos detrás de HOG sin usar el término HOG en una solicitud de patente en 1986. [1] En 1994, los conceptos fueron utilizados por Mitsubishi Electric Research Laboratories . [2] Sin embargo, el uso sólo se generalizó en 2005 cuando Navneet Dalal y Bill Triggs , investigadores del Instituto Nacional Francés de Investigación en Ciencias de la Computación y Automatización (INRIA), presentaron su trabajo complementario sobre descriptores HOG en la Conferencia sobre Visión y Patrones por Computadora. Reconocimiento (CVPR). En este trabajo se centraron en la detección de peatones en imágenes estáticas, aunque desde entonces ampliaron sus pruebas para incluir la detección humana en videos, así como a una variedad de animales y vehículos comunes en imágenes estáticas.



Figura 1.2 representación HOG que captura las principales características de la imagen.

3. Método

La idea esencial detrás del descriptor del HOG es que la apariencia y la forma del objeto local dentro de una imagen se pueden describir mediante la distribución de gradientes de intensidad o direcciones de borde. La imagen se divide en pequeñas regiones conectadas llamados contenedores, y para los píxeles dentro de cada contenedor, se compila un histograma de direcciones de gradiente.

A continuación se mostrarán los pasos esenciales del método:

1. Reajustando el tamaño de la imagen: Independiente del tamaño de la imagen, se utiliza una ventana de detección de 64x128 píxeles, por lo que la imagen se convierte primero en forma (64, 128). Luego, la imagen se divide en partes pequeñas.
2. Calcular los gradientes: necesitamos calcular el gradiente en la dirección x e y. El gradiente son simplemente los pequeños cambios en las direcciones x e y, necesitamos convolucionar dos filtros simples en la imagen. Considerando un bloque de 3x3 píxeles, primero se calcula G_x y G_y para cada píxel. Primero, G_x y G_y se calculan utilizando la fórmula siguiente para cada valor de píxel.

$$G_x(r, c) = I(r, c + 1) - I(r, c - 1) \quad G_y(r, c) = I(r - 1, c) - I(r + 1, c)$$

3. Calcular la magnitud y el ángulo: Después de haber calculado los gradientes, se calcula la magnitud y ángulo para cada pixel utilizando las siguientes fórmulas:

$$Magnitude(\mu) = \sqrt{G_x^2 + G_y^2}$$

$$Angle(\theta) = |\tan^{-1}(G_y/G_x)|$$

4. Después de obtener el gradiente de cada pixel, las de magnitud y ángulo se dividen en celdas de 8x8 para formar un bloque. Para cada bloque, se calcula un histograma de 9 puntos. Un histograma de 9 puntos desarrolla un histograma con 9 contenedores y cada contenedor tiene un rango de ángulo de 20 grados. Cada uno de estos histogramas de 9 puntos se puede trazar como histogramas con contenedores que generan la intensidad del gradiente en ese contenedor. Como un bloque contiene 64 valores diferentes, para los 64 valores de magnitud y gradiente se realiza el siguiente cálculo. Como estamos usando histogramas de 9 puntos, por lo tanto:

$$\begin{aligned} Number\ of\ bins &= 9(ranging\ from\ 0^\circ\ to\ 180^\circ) \\ Step\ size(\Delta\theta) &= 180^\circ / Number\ of\ bins = 20^\circ \end{aligned}$$

Cada J-ésima celda tendrá límites desde:

$$[\Delta\theta \cdot j, \Delta\theta \cdot (j + 1)]$$

Valor del centro de cada celda será:

$$C_j = \Delta\theta(j + 0.5)$$

5. Para cada celda en un bloque, primero calcularemos el j-gésimo contenedor y luego el valor que se proporcionará al j-ésimo y el (j+1)-ésimo contenedor respectivamente. El valor viene dado por las siguientes fórmulas:

$$\begin{aligned} j &= \lfloor \left(\frac{\theta}{\Delta\theta} - \frac{1}{2} \right) \rfloor \\ V_j &= \mu \cdot \left[\frac{\theta}{\Delta\theta} - \frac{1}{2} \right] \\ V_{j+1} &= \mu \cdot \left[\frac{\theta - C_j}{\Delta\theta} \right] \end{aligned}$$

6. Una matriz se toma como un contenedor para un bloque y los valores de V_j y V_{j+1} se agregan a la matriz en el índice del j-ésimo y (j+1)-ésimo contenedor calculados para cada pixel. La matriz resultante después de los cálculos anteriores tendrá la forma de 16x8x9.

7. Una vez que finaliza el cálculo del histograma para todos los bloques, se agrupan 4 bloques de la matriz de histograma de 9 puntos para formar un nuevo bloque (2x2). Este clubbing se realiza de manera superpuesta con un paso de 8 píxeles. Para las 4 celdas en un bloque, concatenamos todos los histogramas de 9 puntos para cada celda constituyente para formar un vector de 36 características.

8. Los valores de f_{bi} para cada bloque están normalizados por la norma L2:

$$f_{bi} \leftarrow \frac{f_{bi}}{\sqrt{\|f_{bi}\|^2 + \varepsilon}}$$

9. Para normalizar, el valor de k se calcula primero mediante las siguientes fórmulas:

$$k = \sqrt{b_1^2 + b_2^2 + b_3^2 + \dots + b_{36}^2}$$

$$f_{bi} = \left[\left(\frac{b_1}{k} \right), \left(\frac{b_2}{k} \right), \left(\frac{b_3}{k} \right), \dots, \left(\frac{b_{36}}{k} \right) \right]$$

10. Esta normalización se realiza para reducir el efecto de los cambios de contraste entre imágenes del mismo objeto. De cada bloque. Se recopila un vector de características de 36 puntos. En la dirección horizontal hay 7 bloques y en la dirección vertical hay 15 bloques. Entonces la longitud total de las características HOG será: $7 \times 15 \times 36 = 3780$. Se obtienen las características HOG de la imagen seleccionada.

Bibliografía:

[1] *Histograms of Oriented Gradients for Human Detection*, Navneet Dalal and Bill Triggs.

<https://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf>

[2] *HOG Feature Extraction in Python*.

<https://towardsdatascience.com/hog-histogram-of-oriented-gradients-67ecd887675f>

[3] *HOG (Histogram of Oriented Gradients): An Overview*.

<https://www.thepythoncode.com/article/hog-feature-extraction-in-python>