

Examen U IV

Angel Manrique Pozos Flores; M07211505
Tecnológico Nacional de México,
Blvd. Industrial, Mesa de Otay, 22430 Tijuana, B.C., México.

3 de mayo de 2016

Explicar la implementación del SIFT en opencv.

- Como realizarías esta implementación?
- Escribir un pseudocódigo.

1. Introducción

La búsqueda de características y puntos de interés en distintas imágenes es un problema común en la visión computacional, cuando dichas imágenes son similares por ejemplo su escala, su orientación, etc. los detectores de esquinas básicos pueden funcionar, pero cuando las imágenes tienen diferentes escalas y rotaciones, se requiere entonces el uso de la transformada de características invariantes a la escala (SIFT) por sus siglas en inglés.

SIFT no solamente es un algoritmo de invariante de escala, también se puede cambiar distintas características obteniendo muy buenos resultados.

- Escala.
- Rotación.
- Iluminación.
- Punto de vista.

2. SIFT

SIFT se compone de 6 partes principales:

- Construcción del espacio de escala - Esta es la preparación inicial, aquí se crean representaciones internas de la imagen original para asegurar que sea invariante a la escala, esto se logra generando un espacio de escalas.
- Aproximación de LoG - El laplaciano de la gaussiana es bueno para encontrar puntos de interés o puntos clave, en una imagen. Pero es costoso computacionalmente, por ello de cierta manera engañamos y utilizamos las representaciones aproximadas que se crearon anteriormente.
- Búsqueda de puntos clave - Con la rápida aproximación, nosotros tratamos de encontrar puntos clave, estos son máximos y mínimos de la diferencia de gaussianas de la imagen que se calcularon previamente.
- Discriminado de puntos de no interés - Bordes y regiones de bajo contraste no son de utilidad como puntos clave, eliminarlos hace que el algoritmo sea mas eficiente y robusto, es una técnica muy similar a la utilizada en el detector de esquinas de Harris.
- Asignación y orientación de los puntos clave - Una orientación es calculada para cada uno de los puntos clave, donde se realizan mas cálculos que sean relativos a esta orientación, esto ayuda a que se cancele de forma efectiva el efecto de la rotación, haciendo con ello que sea invariante a la rotación.
- Generación de las características de SIFT - Finalmente con invariante de la rotación y la escala mas representaciones son generadas, esto ayuda que características únicas puedan ser identificadas

2.1. El espacio de escala

Objetos en el mundo real son significativos solo a cierta escala, por ejemplo si uno observa un árbol busca detalles específicos de la imagen como lo son las hojas, las ramas, de forma intencional.

Conforme se van perdiendo detalles, es necesario el asegurar que no se introducen detalles falsos, la única manera de poder hacer esto es mediante el difuminado o blur gaussiano, por ello para poder crear el espacio de escalas se necesita tomar la imagen original e ir generando progresivamente imágenes difuminadas.



Figura 1: Blur gaussiano aplicado a un set de imágenes.

2.2. Espacio de escalas en SIFT

SIFT toma los espacios de escala del siguiente nivel, uno toma la imagen original y va generando progresivamente imágenes difuminadas, después se cambia de tamaño la imagen original a la mitad de su tamaño y se generan de nueva cuenta imágenes borrosas y así se va repitiendo el proceso.

El numero de octavas y escalas dependerá del tamaño de la imagen original, al utilizar SIFT se necesita declarar cuantas octavas y cuantas escalas se quieren obtener, aunque se sugiere utilizar 4 octavas y 5 niveles de blur.

Matemáticamente el blurring se refiere a la convolucion de un operador gaussiano con la imagen, el blur gaussiano tiene una expresión particular o operador que es aplicado a cada pixel, lo cual resulta en una imagen difuminada.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (1)$$

Donde:

- L es la imagen con blur.
- G es el operador gaussiano de blur.
- I es la imagen.
- x, y son las coordenadas de localización.
- σ es el parámetro de escala, viéndolo de esta forma, entre mas grande el valor mayor el blur.
- El * es el símbolo que representa la operación de la convolucion en x y y, que aplica un blur gaussiano G en la imagen I.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2)$$

Que es el operador de blur gaussiano.

2.3. Aproximación LoG

Utilizando las imágenes que se obtienen del paso anterior se generan nuevas imágenes que son resultado de la diferencia de gaussianas (DoG) por sus siglas en ingles, estas imágenes DoG son utilizadas para la búsqueda de puntos de interés o puntos clave en la imagen.

2.3.1. Laplaciano del gaussiano

El laplaciano del gaussiano (LoG) es una operación que toma una imagen y se procede a aplicar un blur gaussiano, después se calcula la segunda derivada o el laplaciano, con ello se localizan los bordes y contornos de la imagen, estos bordes y esquinas son de utilidad para poder localizar puntos de interés.

Pero una segunda derivada es extremadamente sensible al ruido, por ello el blur gaussiano suaviza el ruido y estabiliza la segunda derivada, el problema surge al calcular todas estas segundas derivadas ya que tienen un alto costo computacional por ello se utiliza esta aproximación.

Para generar imágenes de las laplacianas de las imágenes gaussianas de forma rápida, se usa el espacio de escalas, donde se calcula la diferencia entre dos escalas consecutivas o la diferencia de la las gaussianas como se observa en la figura 3.

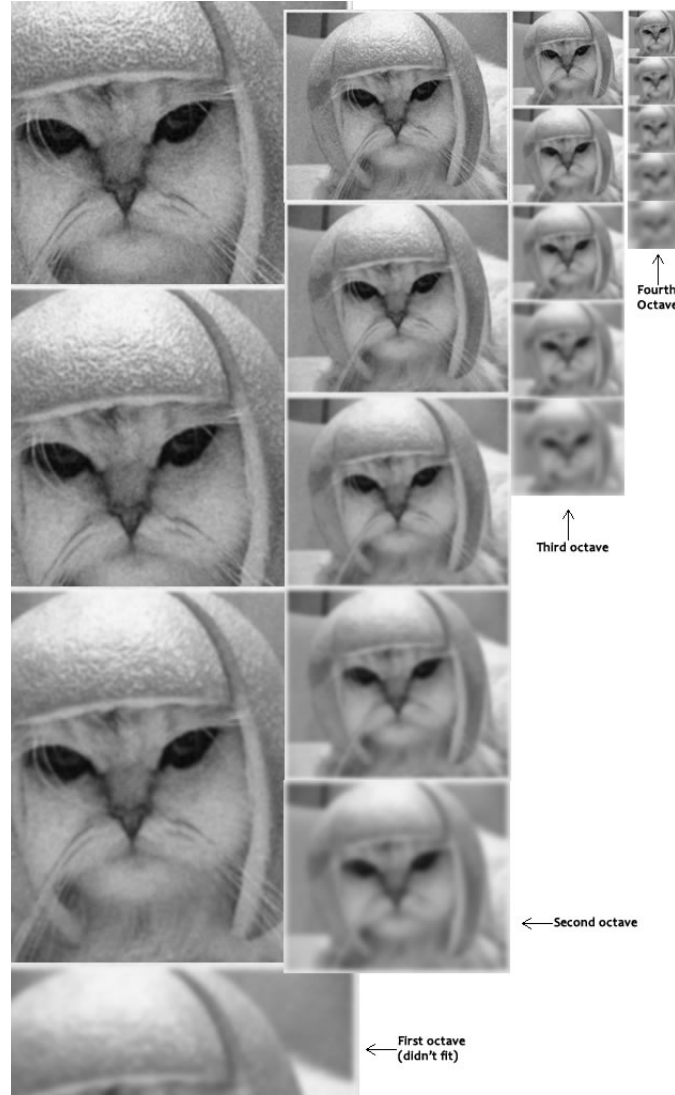


Figura 2: SIFT espacio de escalas.

Estas diferencias de gaussianas son equivalentes al laplaciano de la gaussiana y con ello se puede reemplazar un proceso que es costoso computacionalmente con una simple resta, estas imágenes DoG traen también una característica muy importante, que son invariantes a la escala.

El laplaciano de las imágenes gaussianas no es bueno, ya que no es invariante a la escala, ya que depende de la cantidad de blur que se defina donde la expresión para la gaussiana.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (3)$$

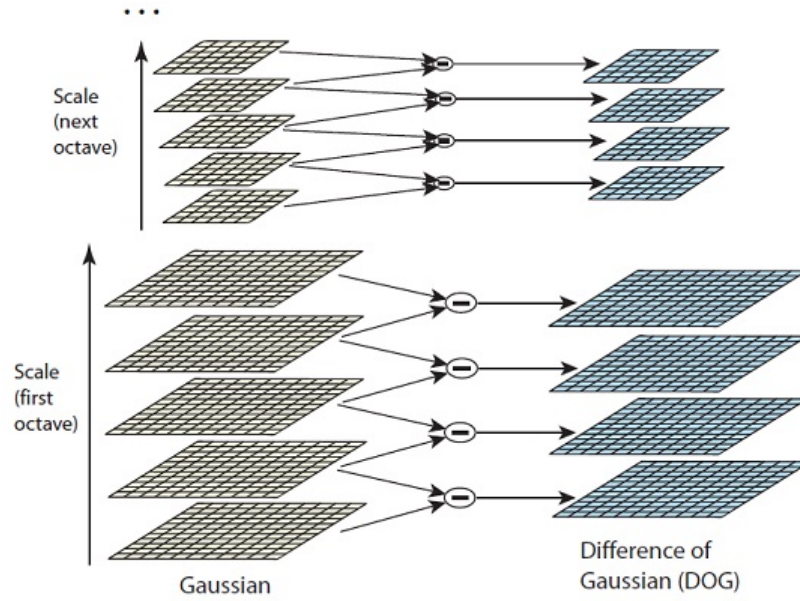


Figura 3: Diferencia de gaussianas (DoG).

Si observamos el termino σ^2 hace referencia a la escala, y el laplaciano de la gaussiana de representa de la siguiente manera:

$$\nabla^2 G \quad (4)$$

Entonces el laplaciano del gaussiano invariante a la escala quedaría.

$$\sigma^2 \nabla^2 G \quad (5)$$

Donde se evitan todas estas complejidades al utilizar los DoG, donde la imagen obtenida de la operación ya esta multiplicada por σ^2 . Esta operación se realiza con todas las octavas, lo cual va a generar DoG de múltiples tamaños.

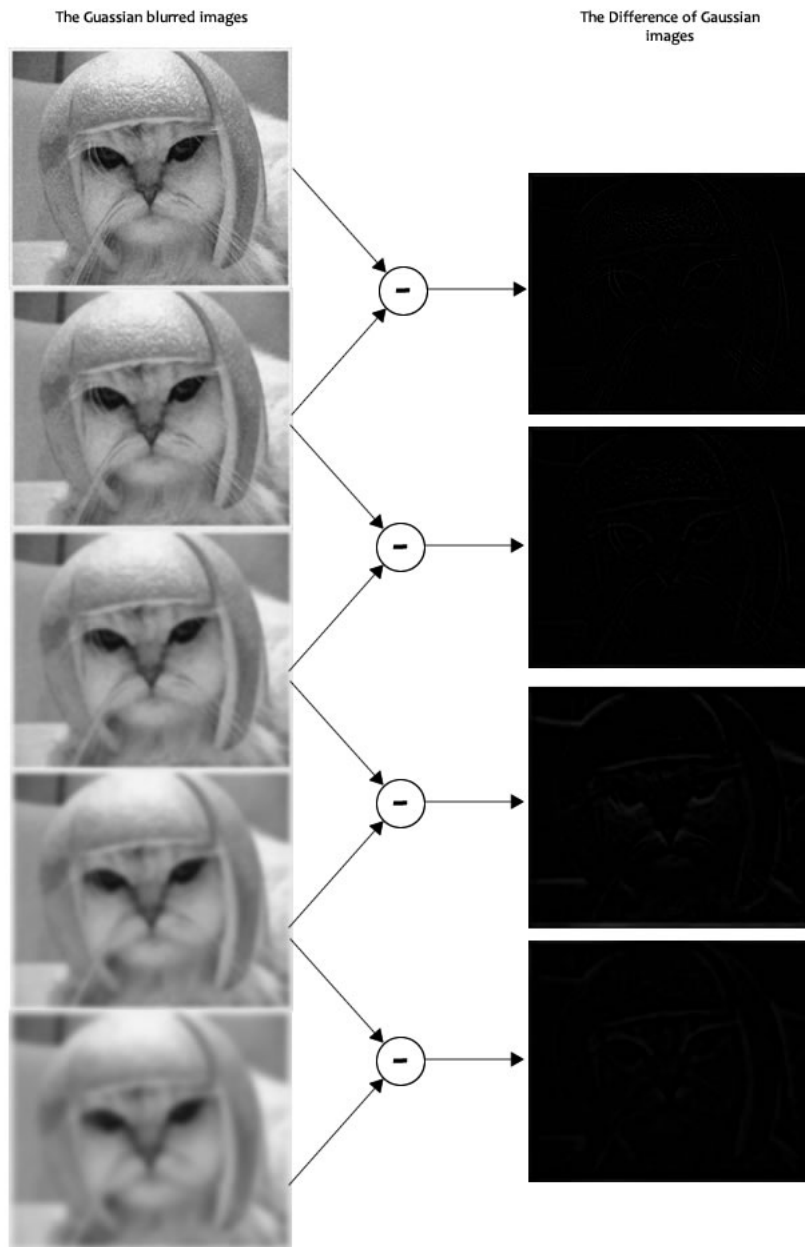


Figura 4: Diferencia de gaussianas aplicada a una imagen.

2.4. Búsqueda de Keypoints

Aquí se utiliza una aproximación como la que se hace en el detector de esquinas de Harris, donde se checan las intensidades y se realiza una discriminación.

Donde si la magnitud de la intensidad en el pixel actual en la imagen DoG es menor que cierto valor umbral, entonces es rechazado.

Debido a que tenemos puntos de interés a nivel de subpixel se utiliza la expansión de Taylor para obtener el valor de intensidad en ciertas locaciones de nivel subpixel, donde, si su magnitud es menor que cierto valor de intensidad, entonces se rechaza este punto de interés.

2.4.1. Localización de los máximos y mínimos

Primero se procede a encontrar el valor máximo y mínimo, esto resulta simple, se itera con cada pixel y con los pixeles del vecindario, este termina cuando se obtiene el punto de interés por ejemplo.

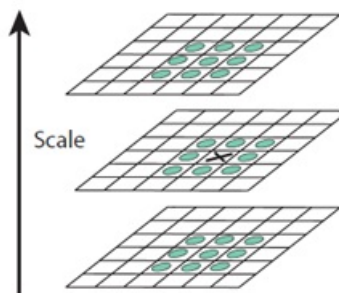


Figura 5: La x marca el pixel actual, los círculos verdes corresponden a los pixeles vecinos, donde x es marcado como el punto de interés o punto clave si es el pixel que contiene el mayor valor de todos los vecinos alrededor de la imagen.

Una vez realizado esto los puntos marcados son los valores aproximados del máximo y el mínimo correspondientes, como se muestra en la figura 6.

2.5. Removiendo bordes de bajo contraste

La idea es calcular dos gradientes en un punto de interés, uno perpendicular al otro y basado en la imagen alrededor del punto de interés, existirán 3 posibilidades:

- Una región plana- En este caso ambos gradientes son muy pequeños.
- Un borde - Aquí un gradiente es grande perpendicular a la esquina y el otro es pequeño, a lo largo de la esquina.

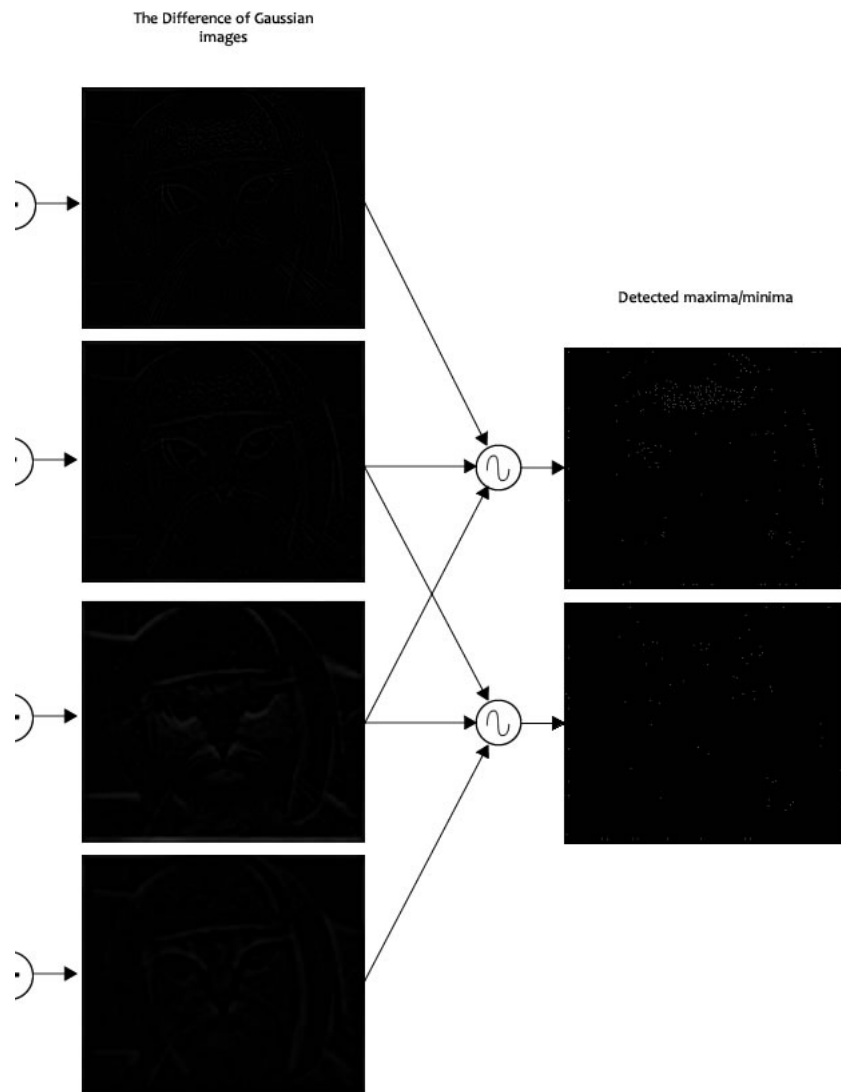


Figura 6: Detección de los máximos y mínimos.

- Una esquina - Aquí ambos gradientes tienen un valor grande.

Las esquinas son buenos puntos de interés, por ello solo nos interesan las esquinas, si ambos gradientes son lo suficientemente grandes, se tomaran como un punto de interés de lo contrario serán rechazados.

Esto se logra mediante la matriz Hessiana, usando esta matriz uno puede localizar fácilmente si es una esquina o no, por ejemplo:

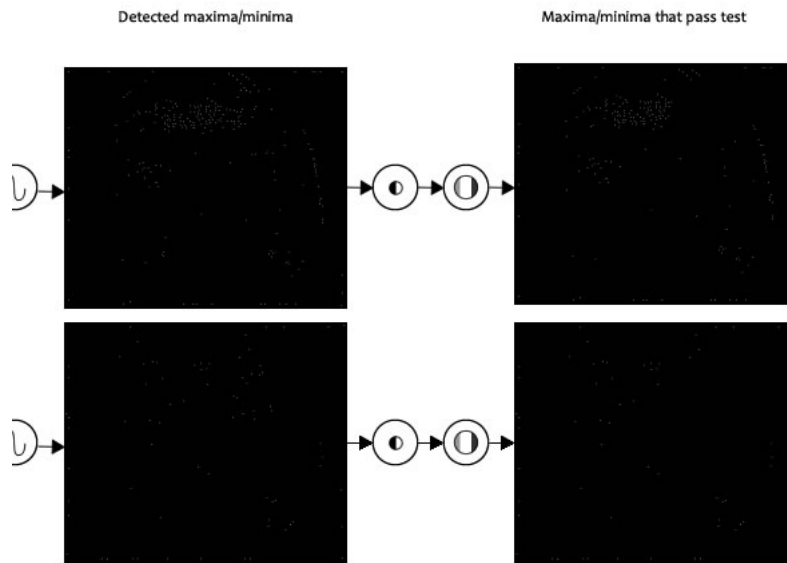


Figura 7: Aquí se hacen dos test, uno que corresponde a los niveles de contraste y el otro a los bordes, donde se obtienen los puntos de interés con mayor peso.

2.6. Orientación de los puntos de interés

Una vez que tenemos los puntos de interés, se debe buscar su estabilidad, sabemos que son estables a la invariante de escala, por ello, el siguiente paso es asignar una orientación a cada punto de interés, esta orientación nos dará una invariante a la rotación, donde mayor sea el valor de invariancia mejor sera.

La idea es coleccionar las direcciones del gradiente y las magnitudes alrededor de cada punto de interés, entonces podemos definir la orientación mas importante de esa región y asignar con ello la orientación para ese punto de interés.

Estas magnitudes y direcciones se calculan utilizando las siguiente ecuación.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (6)$$

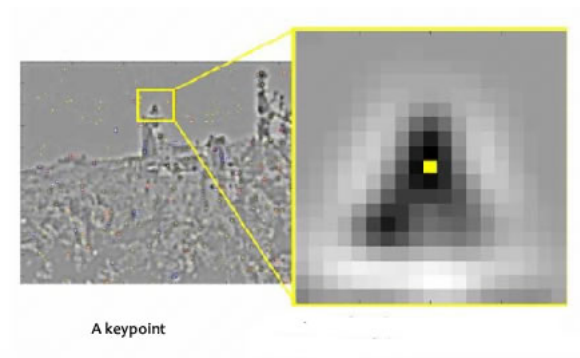


Figura 8: El tamaño de la orientación de la región alrededor del punto de interés depende de la escala, entre mayor sea la escala, mayor sera la región de colección.

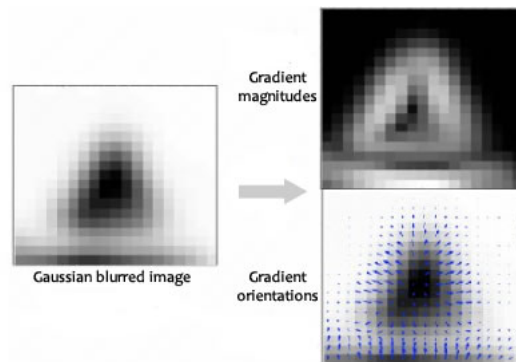


Figura 9: Aquí se muestran las magnitudes y orientaciones de los gradientes del punto de interés.

Después la magnitud y la orientación es calculada para todos los pixeles alrededor del punto de interés, de aquí se obtiene un histograma, en el histograma los 360 grados de orientación son repartidos en 36 segmentos, uno cada 10 grados, donde las cantidades corresponden a una cantidad proporcional a la magnitud y el gradiente en ese punto.

2.7. Generación de características

Como ultimo paso, se sabe que es invariante a la escala y a la rotación ahora se procede a crear una marca para cada punto de interés, con el fin de poder identificarlos.

Esto se logra utilizando una función gaussiana ponderada, esta función nos genera un gradiente que se multiplica con la magnitud de las orientaciones y ob-

tienes de esta manera algo ponderado, entre mas grande el valor, menor sera la magnitud.

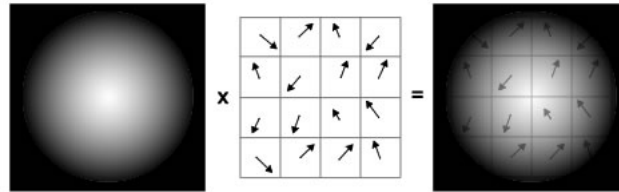


Figura 10: Aquí nuestro punto de interés no esta exactamente en un pixel, por ello se debe interpolar la imagen para poder generar la orientación y la magnitud de los pixeles.

Este vector característico introduce algunas complicaciones.

- Dependencia a la rotación - El vector característico usa las orientaciones de los gradientes, donde si rotamos la imagen todo cambiara, para lograr esta independencia a la rotación los puntos de interés son substraídos de cada orientación, entonces cada posición de los gradientes sera relativa a la posición del punto de interés.
- Dependencia a la iluminación - Se hace una aproximación a cierto nivel de thresholding y se normaliza el vector resultante para poder hacerlo independiente a la iluminación.

3. Conclusiones

Primero generaría varias octavas de la imagen original, el tamaño de cada una de las octavas debe corresponder a la mitad de la imagen previa, conforme cada octava, las imágenes se irán difuminando progresivamente con el uso del operador de gaussiano de blur, después dos imágenes consecutivas de una octava son tomadas y una es restada a la otra, se procede a tomar el siguiente par y se repite el proceso, el resultado son imágenes que se aproximan a laplaciano del gaussiano de escala invariante.

Los puntos de interés se reducen, con esto se incrementa la eficiencia y por lo tanto la robustez del algoritmo. Para asignar una orientación se utiliza un histograma y una pequeña región que lo rodea, mediante el histograma podemos localizar la orientación más prominente del gradiente, donde si solo es un pico este es asignado al punto de interés, después se crea un vector característico que contiene la información de la magnitud y dirección del punto de interés para poder ser utilizado.

El pseudo-código quedaría de la siguiente manera:

1. Se toma una imagen y se convierte a escala de grises.
2. Declaramos la cantidad de escalas y octavas a aplicar.
3. Se aplican distintos valores de blur gaussiano a la imagen original, generando múltiples versiones con blur de la misma imagen.
4. Se genera el espacio de escalas, tomando la imagen original a la mitad de su tamaño y se le aplica el proceso anterior aplicando un set de distintos valores de blur gaussiano.
5. Ahora procedemos a generar la aproximación de las DoG, tomando la diferencia entre imágenes con gaussiana aplicada, con ello podemos hacer la búsqueda de los puntos de interés.
6. Aquí discriminamos con un método parecido al de Harris, donde se checan las intensidades y se va realizando una discriminación, donde si la magnitud de la intensidad en el pixel actual en la imagen DoG es menor que cierto valor umbral, entonces es rechazado.
7. Se localizan ahora los valores de los máximos y los mínimos, donde las esquinas son los puntos de interés utilizados, aquí se utiliza la matriz Hessiana para encontrar dichos puntos .

8. Una vez localizados, buscaría que fueran estables o invariantes a la rotación.
9. Calcularía entonces la magnitud y dirección de los píxeles alrededor del punto de interés y obtendría un histograma característico de la zona que encierra a mi punto de interés.
10. Tomaría entonces la orientación que corresponde al pico mas grande del histograma y este lo asociaría con mi punto de interés.
11. Ya sabemos que es el punto de interés es invariante a la escala debido a la aproximación por las DoG y también invariante a la rotación, ahora buscaría una forma de etiquetarlos para poder identificarlos, esto lo haría con una gaussiana ponderada, también buscaría el nivel adecuado de threshold y lo normalizaría para que fuera independiente de la iluminación.

4. Bibliografía

- The Scale Invariant Feature Transform (SIFT) .
(<http://tinyurl.com/zfu8m5r>)
- Introduction to SIFT
(<http://tinyurl.com/zen8bsn>)
- SIFT.
(<http://tinyurl.com/heutewl>)