

Поиск светлых пятен

1 Задача:

Рассматриваем задачу обработки изображений. Требуется на монохромном изображении выделить светлые пятна - однородные яркие участки с резким изменением градиента яркости на границе. Ключевые слова для подобных алгоритмов - *blob detection*.

В данной задаче под изображением понимаем матрицу $Z = \{z_{ij}\}, i = 1, \dots, m, j = 1, \dots, n$ где $z_{ij} \in B$ - яркость пиксела. Здесь $B \in \mathbb{R}_+$ - возможные значения яркости. В компьютерной графике обычно $B = \{1, \dots, 256\}$.

2 Алгоритм:

2.1 Описание:

Алгоритм должен получать на вход изображение Z и, возможно, набор параметров. На выходе требуется получить бинарную матрицу R в которой единицам соответствуют центры найденных пятен.

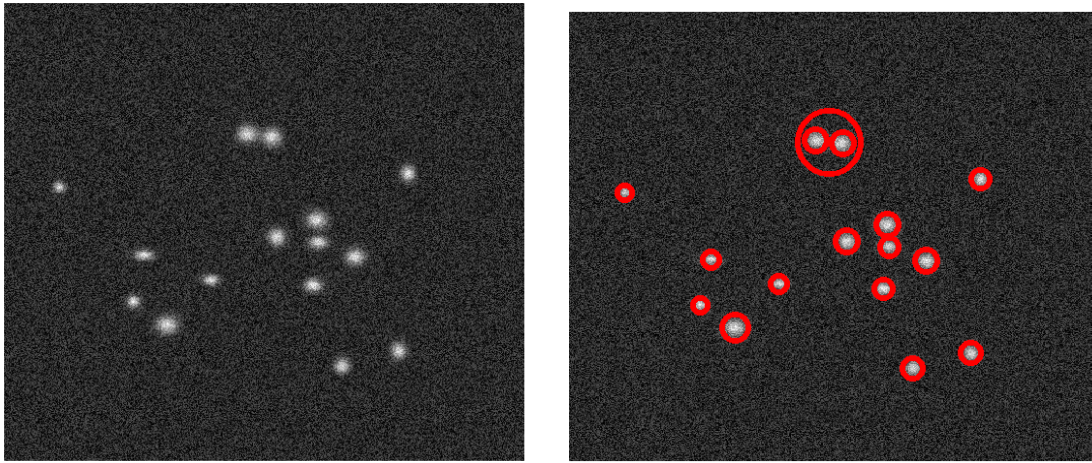


Рис. 1: Пример работы алгоритма.

Для нахождения светлых пятен воспользуемся одним из самых известных алгоритмов в данной области LoG-детектор [1] (Laplacian-of-Gaussian). В этом алгоритме производится свертка изображения со второй производной гауссианы и затем на получившемся изображении ищутся локальные минимумы. Они будут соответствовать светлым пятнам на изображении.

В предлагаемом алгоритме для поиска пятен различного размера вычисляется множество свертки изображения с масками с различным параметром. На всех свертках точки, соответствующие центрам ярких пятен, будут являться локальными минимумами. При этом минимальное значение яркости будет достигаться на свертке с маской с параметром равным радиусу светлого пятна.

Для устойчивости к шуму предлагается искать локальные минимумы минимумы на изображении \overline{conv} , которое определяется как $\overline{conv}(i, j) = \min_{k=1, \dots, k} (conv_k(i, j))$.

А также среди найденных пятен отсекаются те, интенсивность которых близка к интенсивности шумов.

2.2 Алгоритм:

Обозначения:

- Z - входное изображение

- R - матрица, совпадающая по размеру с Z , с элементами $r_{ij} \in \{0, 1\}$, в которой единицы соответствуют центрам ярких пятен.
- $\mathbf{r} = (r_1, r_2, \dots, r_k)$ - набор возможных радиусов ярких пятен среди которых ведется поиск
- $G(x, y, \sigma)$ - гауссиана с центром в (x, y) и параметром σ .
- $\hat{G}(x, y, \sigma)$ - нормированная гауссиана: $\hat{G}(x, y, \sigma) = \sigma^2 G(x, y, \sigma)$
- $\nabla \hat{G}(x, y, \sigma)$ - вторая производная нормированной гауссианы
- $Convolution(img_1, img_2)$ - функция свертки изображений img_1 и img_2
- $\mathbf{Conv} = (conv_1, \dots, conv_k)$ - множество сверток.

Вход: Z, \mathbf{r}

Выход: R

```

1:  $R \leftarrow 0^{m \times n}$ 
2: // создаем набор сверток с различными масками
3: для всех  $r_k$  в  $\mathbf{r}$ 
4:    $conv_k \leftarrow Convolution(Z, \nabla \hat{G}(x, y, r_k))$ 
5:  $\overline{conv} \leftarrow \min(conv_1, \dots, conv_k)$ 
6: для всех  $(i, j)$  в  $[1, \dots, m] \times [1, \dots, n]$ 
7:   если  $(i, j)$  - точка локального минимума в  $\overline{conv}$  то
8:      $R_{ij} \leftarrow 1$ 
9: // отбрасываем шумовые пятна
10: для всех  $(i, j)$  - центры пятен
11:   если  $\overline{conv}(i, j) > \text{median}(\overline{conv}) + 0.1 \left( \max_{(p,q) \in \text{blobs\_centers}} \overline{conv}(p, q) - \text{median}(\overline{conv}) \right)$  то
12:      $R(i, j) = 0$ 

```

2.3 Пошаговые примеры:

Пошаговые иллюстрации работы алгоритма на примере изображения с одним ярким пятном в центре и шумом интенсивностью 0.3 яркости пятна.

1. Изображение

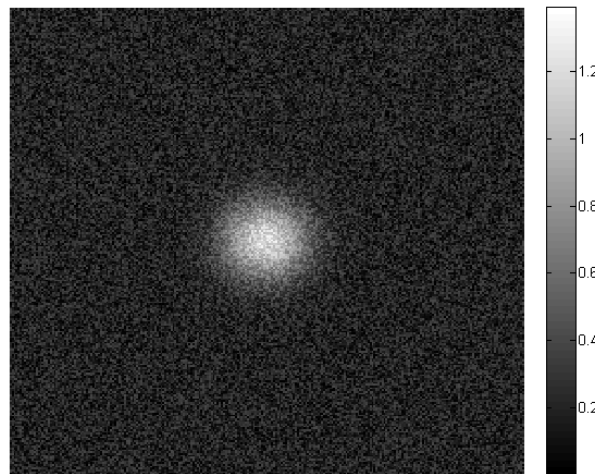


Рис. 2: Исходное изображение

2. Затем вычисляются свертки изображения с масками с различными параметрами:

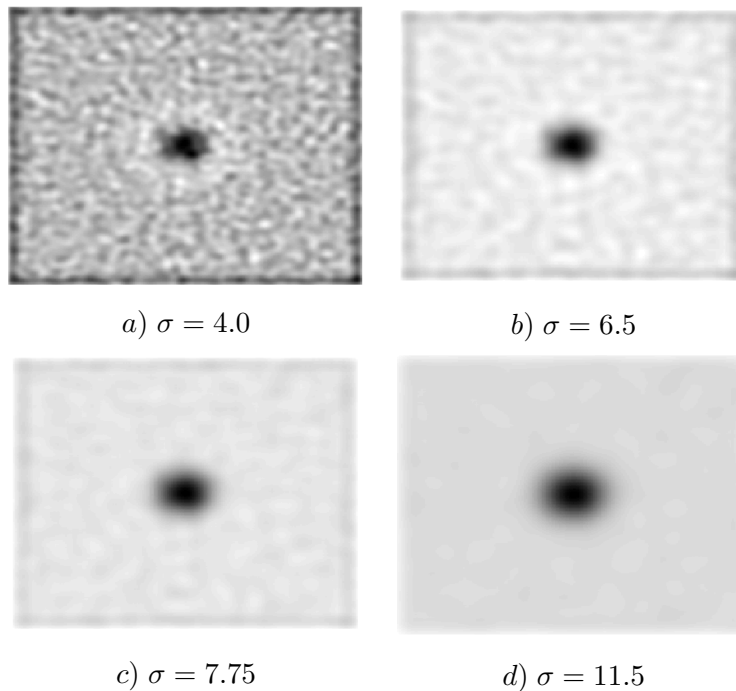


Рис. 3: Свертки с различными масками.

3. Вычисляется изображение - $\overline{conv}(i, j) = \min_{k=1, \dots, k} (conv_k(i, j))$.

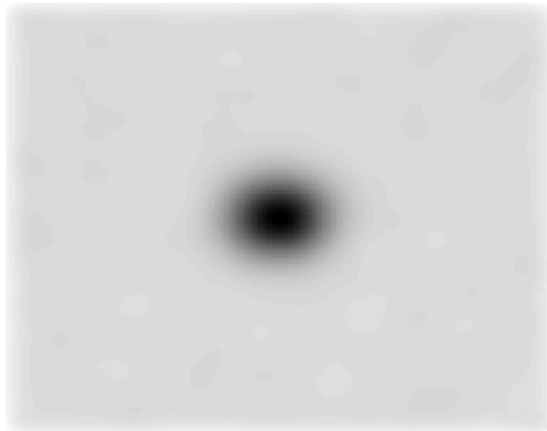


Рис. 4: $\overline{conv}(i, j)$

4. Ищутся локальные минимумы на этом изображении.

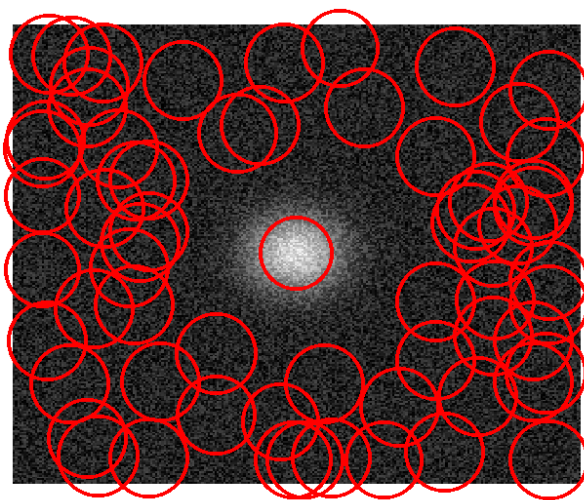


Рис. 5: Шумовые яркие пятна

5. Отфильтровываются шумовые пятна.

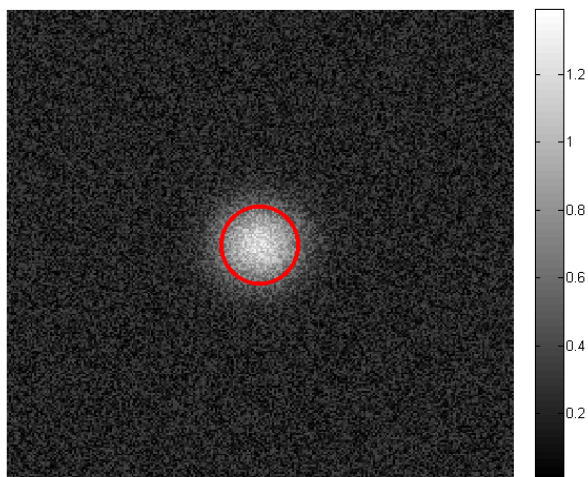


Рис. 6: $\overline{conv}(i, j)$

Список литературы

- [1] D. Marr and E.C. Hildreth. Theory of edge detection. Proc. Roy. Soc. London., B-207:187–217, 1980.
- [2] Krystian Mikolajczyk and Cordelia Schmid, Scale and affine invariant interest point detectors, *International Journal of Computer Vision*, pp. 63–86, 2004