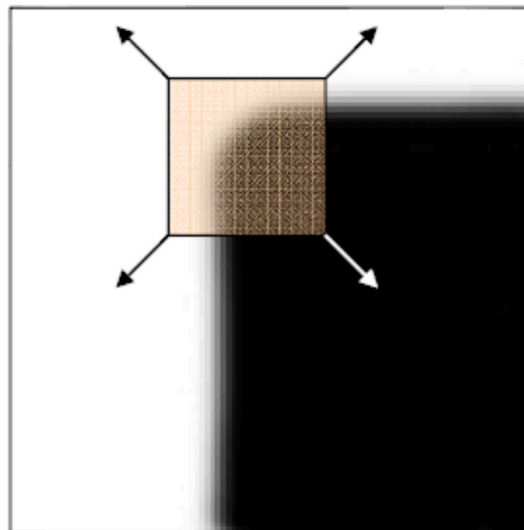


# **Применение детектора Харриса для поиска углов**

# Принцип работы

- Угол - точка пересечения двух прямых
- В окрестности такой точки градиенты сильно разные (в каких то направлениях большие, в каких-то маленькие), по этому признаку ее и ищем



# Определение изменения градиентов

- После продолжительной математической работы получаем

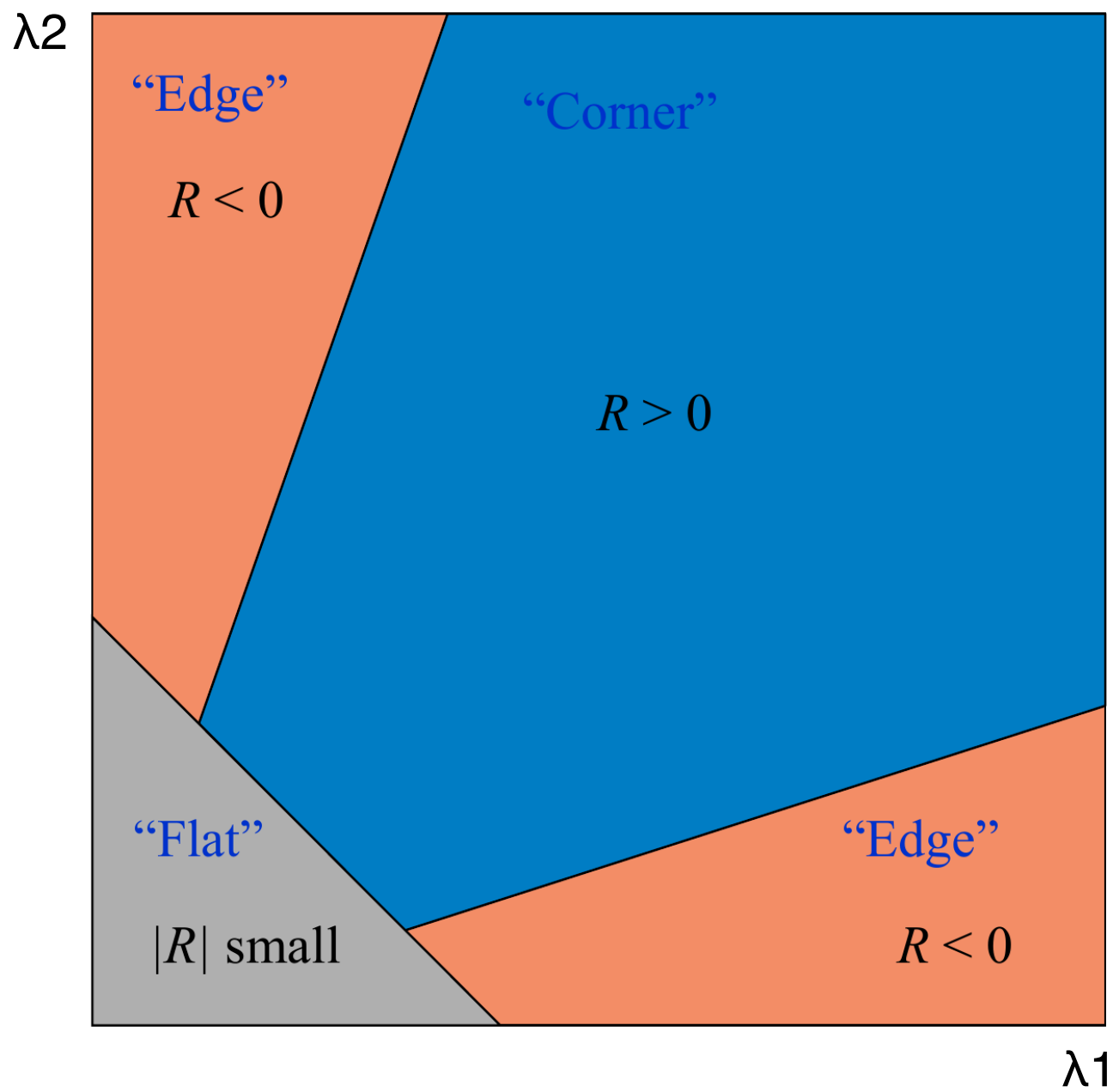
$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

$$E(u, v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

- Откуда для каждой точки можно вычислить величину

$$R = \det(M) - k(\text{trace}(M))^2$$

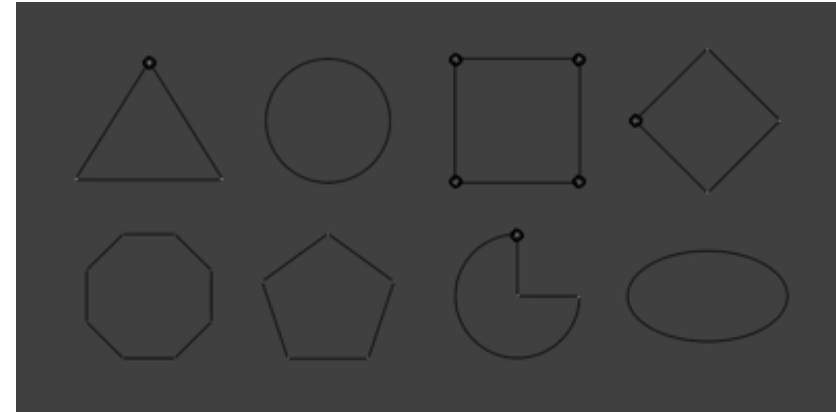
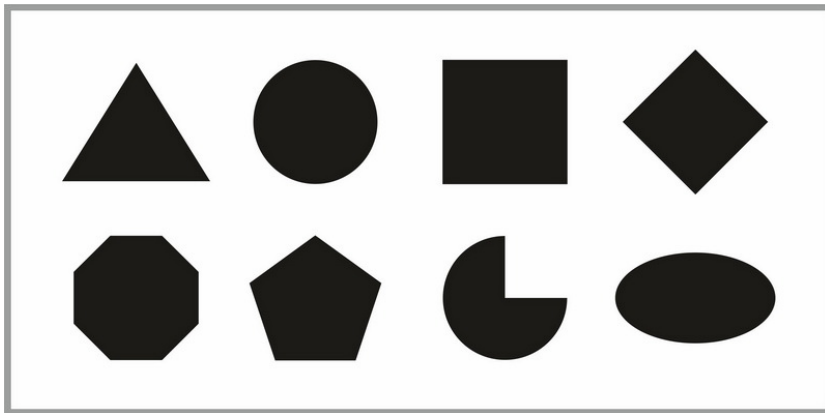
- Если ее значение велико, значит оба собственных значения матрицы  $M$  в этой точке велики, что означает что эта точка - угол



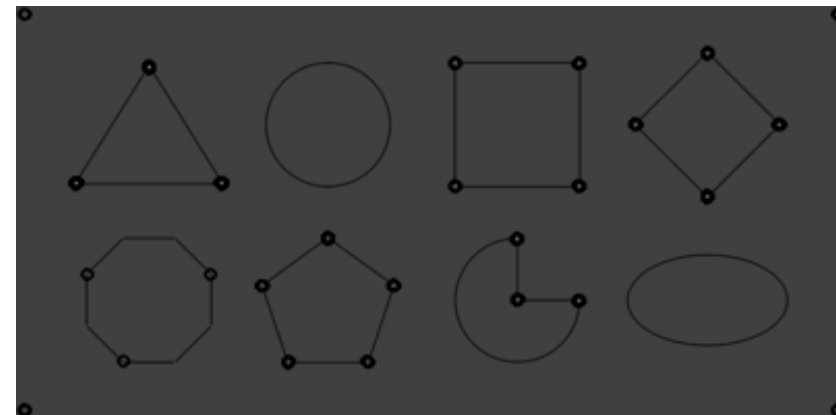
# Применение детектора

1. Переводим изображение в градации серого, чтобы не мучаться с разными каналами
2. Применяем детектор Харриса
3. Нормализуем полученные значения (в нашем случае растягивали на отрезок 0..255)
4. Выделяем те точки, в которых значения  $R$  превысили некоторый порог

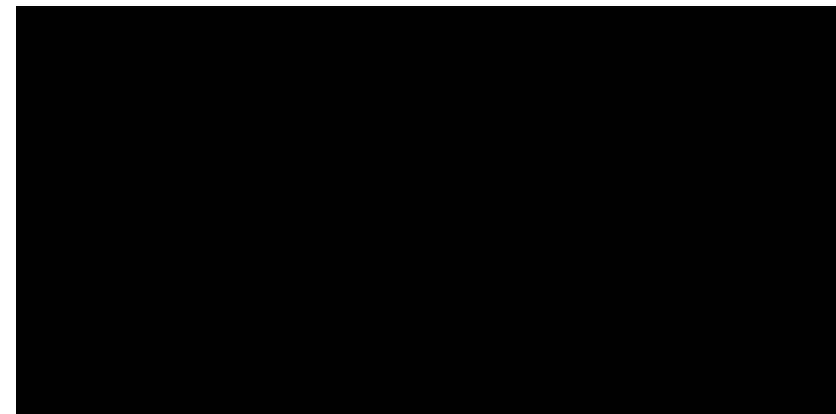
# binar1.jpg



200



64

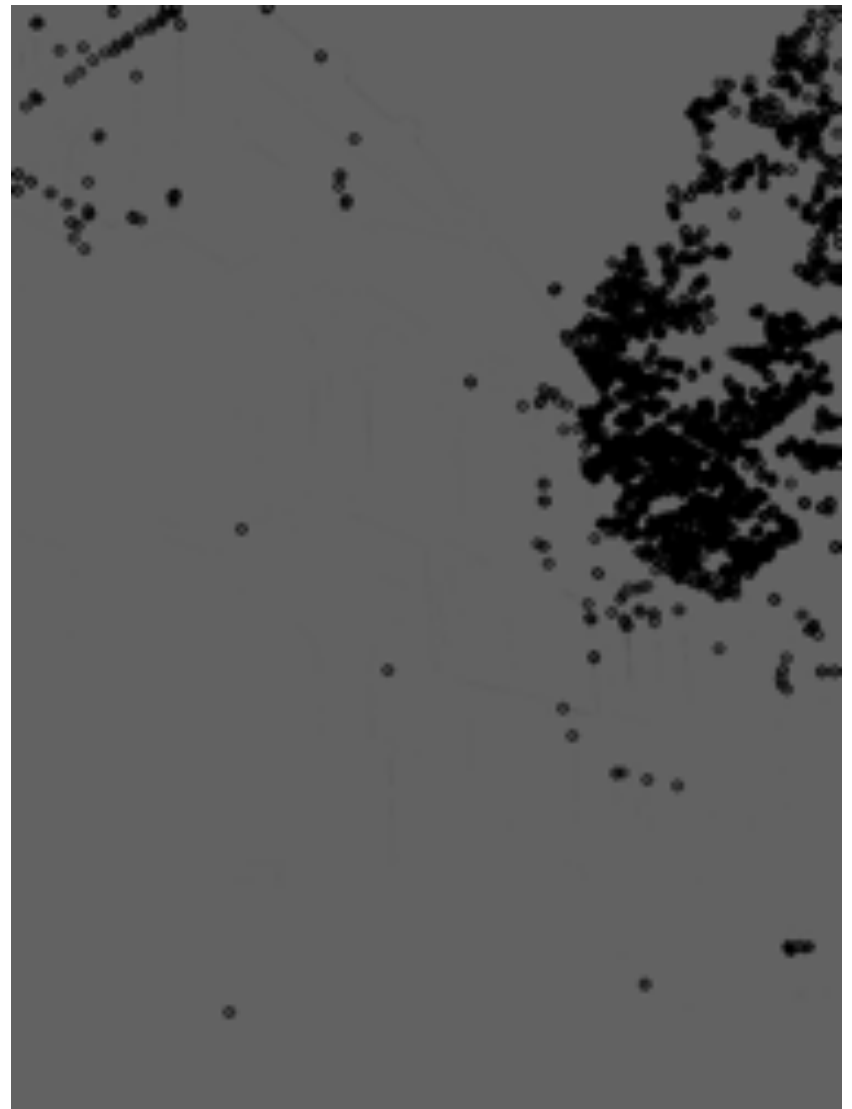


63

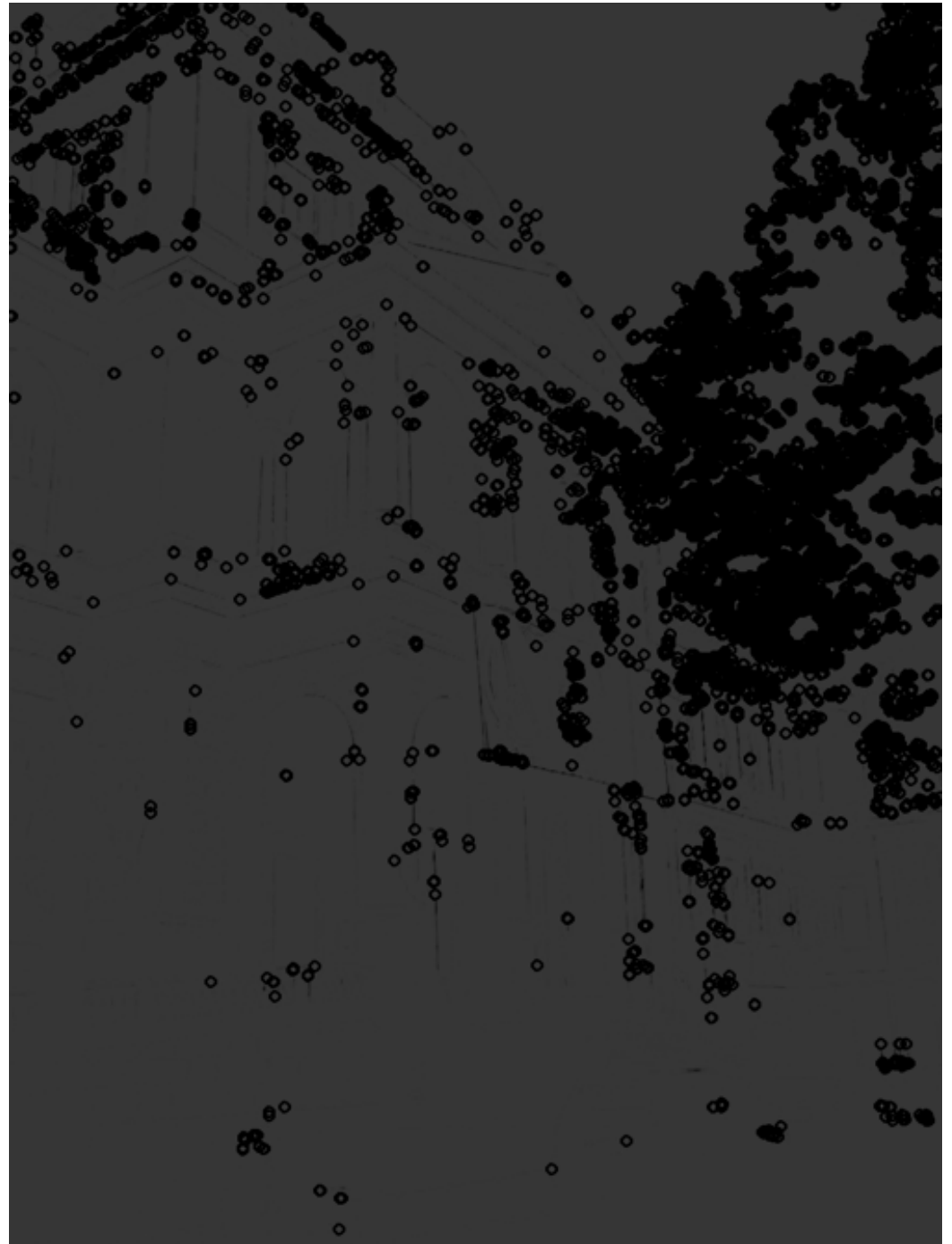
# Размытые изображения



UW1.jpg



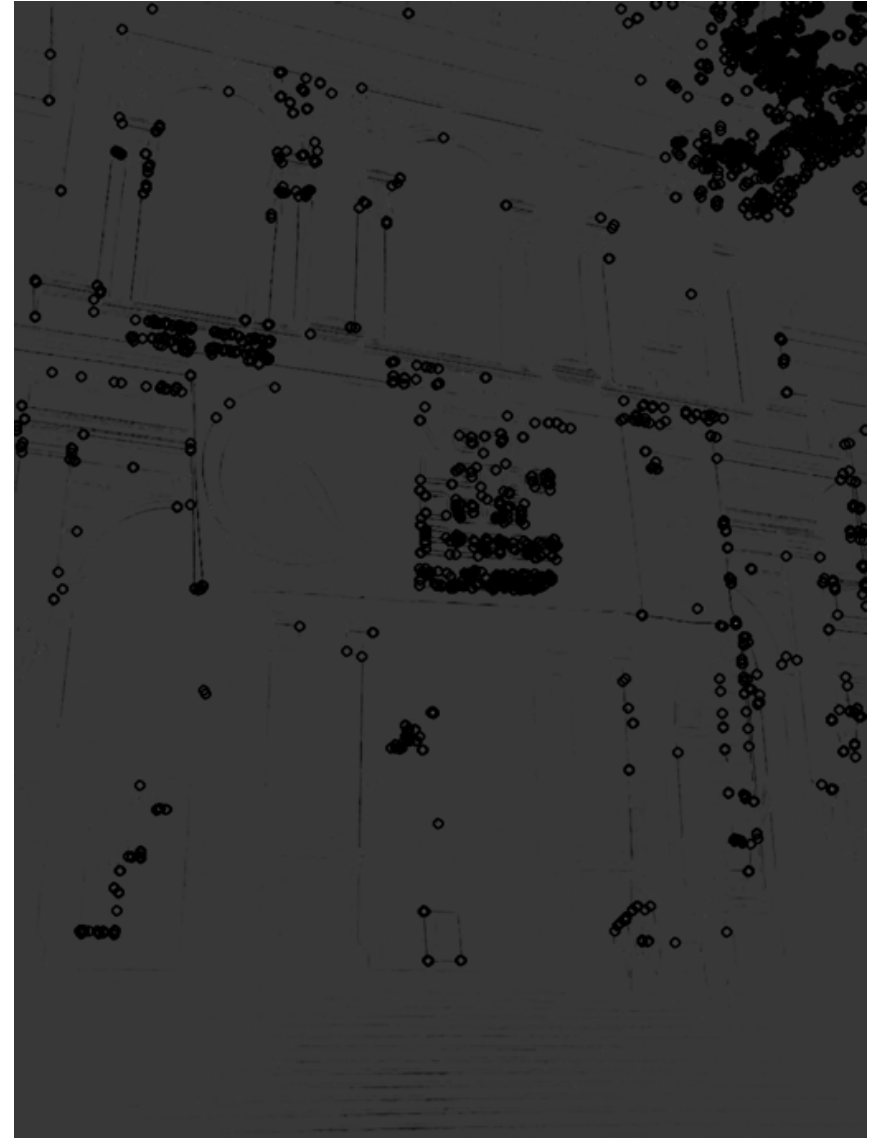
```
sharp_kernel = np.array([
    [-1, -1, -1],
    [-1, 9, -1],
    [-1, -1, -1]
])
src_gray = cv.filter2D(src_gray, -1, sharp_kernel)
```





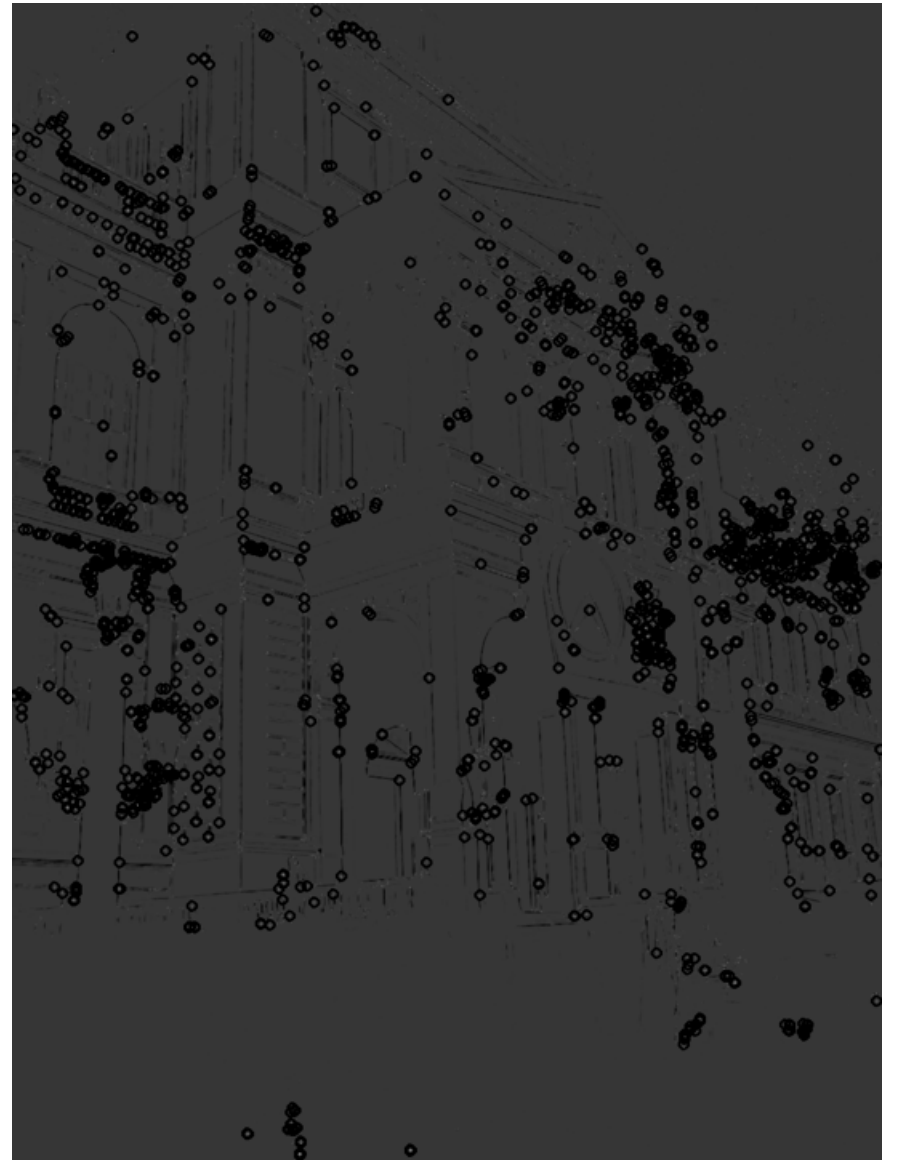


UW2.jpg





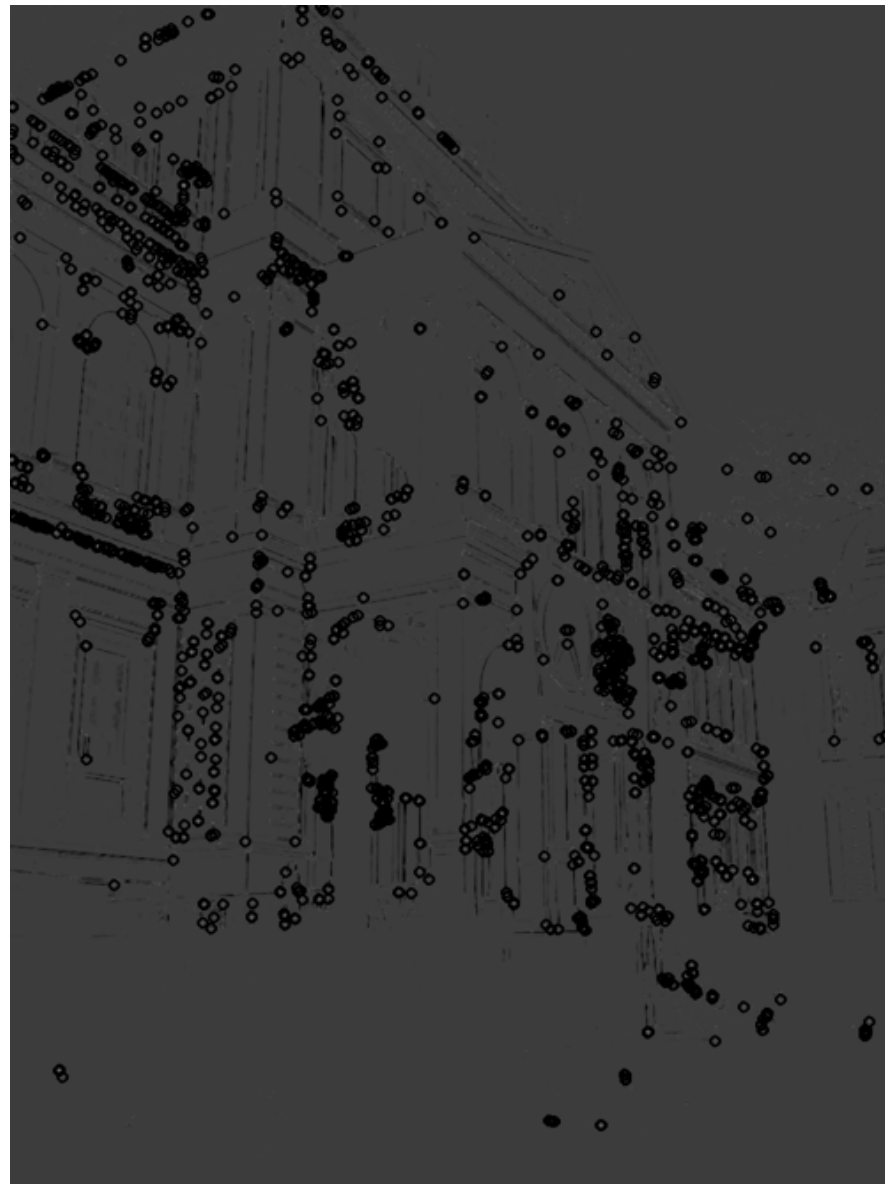
UW3.jpg







UW5.jpg



100



Zurich-49.jpg

