Лекция 8
Суперразрешение изображений

Насонов Андрей Владимирович

http://imaging.cs.msu.ru/

Laboratory of Mathematics Methods of Image Processing
Department of Computational Mathematics and Cybernetics
Lomonosov Moscow State University

Задача восстановления изображений

- Общая математическая модель:

$$u = Az + n$$

Наблюдаемое изображение

Оператор, искажающий изображение

Реальное изображение

Шум

- Требуется найти z при известном u
- Задача является некорректно поставленной

Задача восстановления изображений

- Некорректно поставленная задача

$$Az = u$$

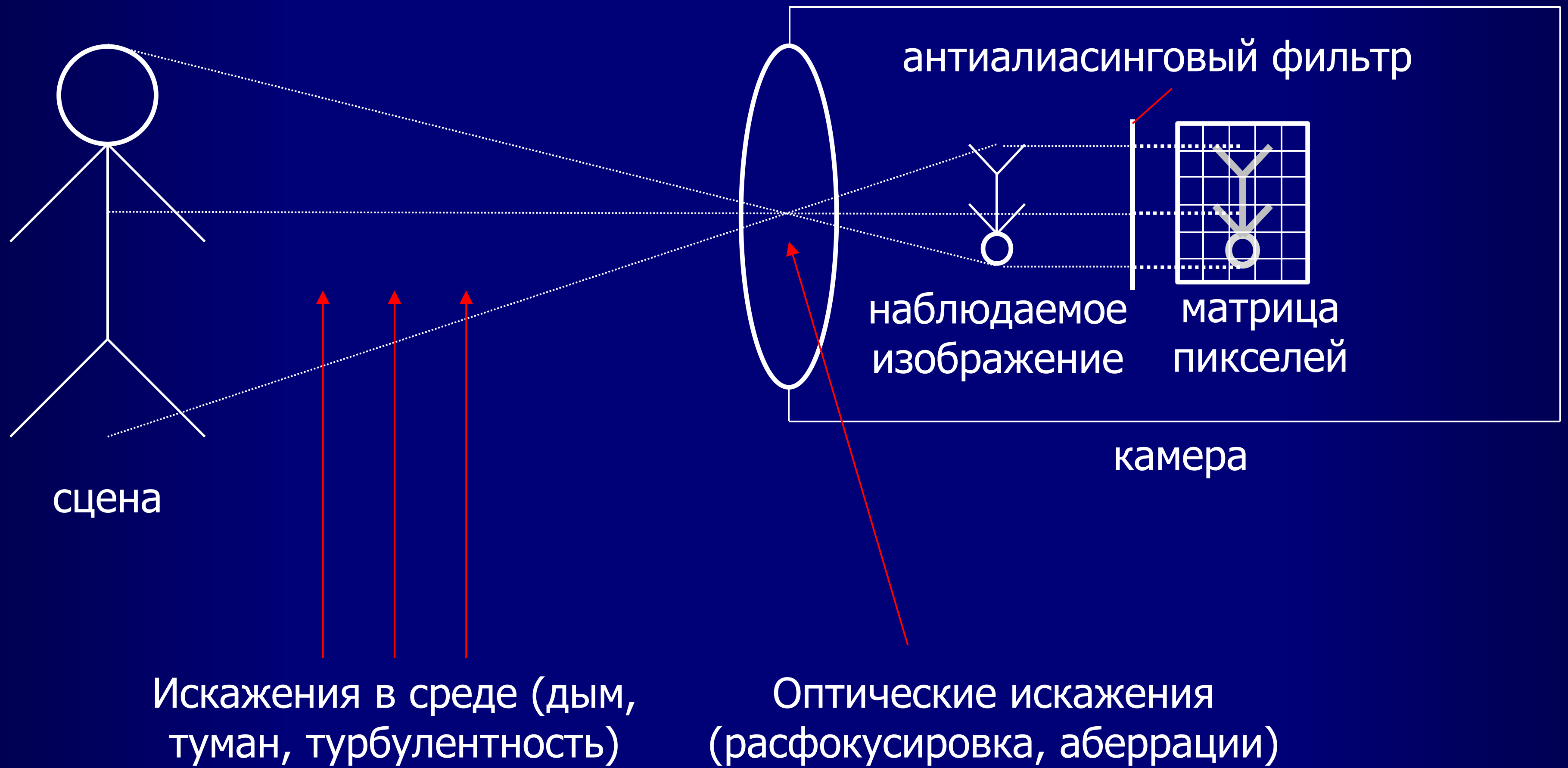
  - Решение не существует
  - Решение не является единственным
  - Неустойчивость
- Метод регуляризации Тихонова

$$z_\alpha = \arg\min_z (\|Az - u\| + \alpha\Omega[z])$$

  - с одной стороны, ищется z такое, чтобы оно соответствовало наблюдаемому изображению u
  - с другой стороны, минимизируется штрафная функция

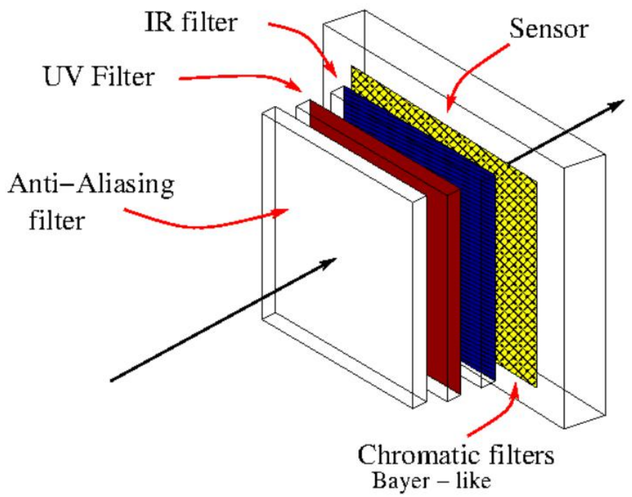
Повышение разрешения изображений

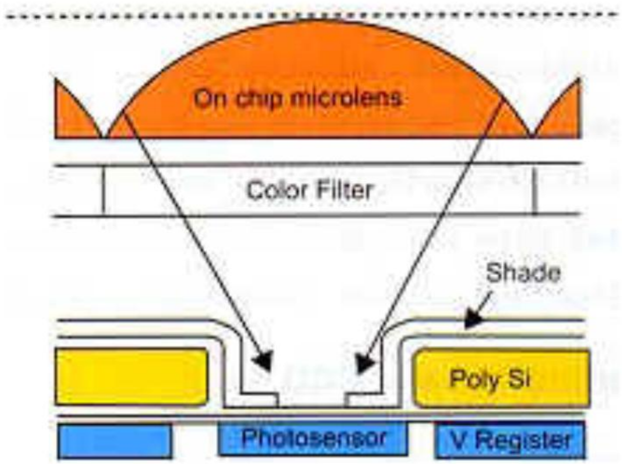
- Математическая модель



Повышение разрешения изображений

- Модель PSF камеры





- Хорошее приближение – фильтр Гаусса с параметром σ, прямо пропорциональным h:

$$\sigma = \sigma_0 h$$

  – На практике используются значения $\sigma_0 \in [0.3,\ 0.35]$

Повышение разрешения изображений

- Математическая модель

$$z = D_h H_h f$$

  - $f(x, y)$ – наблюдаемое изображение
  - h – шаг дискретизации
  - H_h – антиалиасинговый фильтр + PSF камеры
  - D_h – оператор дискретизации

$$z = D_h f \Leftrightarrow z_{i,j} = f(ih, jh)$$

Повышение разрешения изображений

- Задача
  - Дано z_{h_1} – дискретизованное изображение для камеры с шагом h_1 для некоторого наблюдаемого изображения $f(x, y)$
  - Требуется построить соответствующее ему дискретизованное изображение z_{h_2}, которое бы было получено, если бы шаг дискретизации был h_2

Повышение разрешения изображений

- Задача
  - Дано z_{h_1} – дискретизованное изображение для камеры с шагом h_1 для некоторого наблюдаемого изображения $f(x, y)$
  - Требуется построить соответствующее ему дискретизованное изображение z_{h_2}, которое бы было получено, если бы шаг дискретизации был h_2

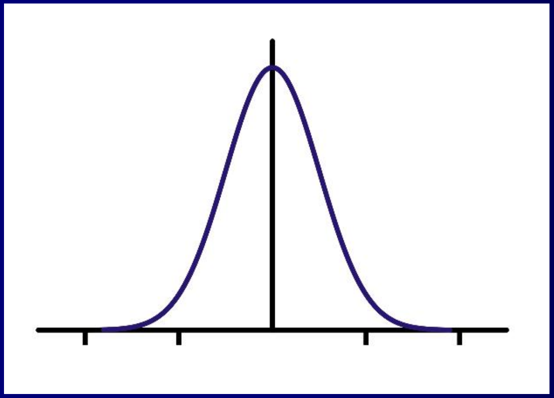
$$z_{h_2} = D_{h_2} H_{h_2} H_{h_1}^{-1} D_{h_1}^{-1} z_{h_1}$$

  - Задача является некорректно поставленной

Повышение разрешения изображений

- Фильтр Гаусса

$$G_\sigma = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$



- Свойства
  - 1. $G_{\sigma_1} * G_{\sigma_2} = G_{\sqrt{\sigma_1^2+\sigma_2^2}}$
  - 2. $\frac{1}{\mathrm{h}^2}\sum_{i,j} G(ih, jh) \approx 1\ \forall\, h > 0$

Повышение разрешения изображений

- Частный случай: задача понижения разрешения изображений в s раз: $h_1 = h$, $h_2 = hs$, $s \geq 1$

$$z_h = D_h\left(f * G_{\sigma_0 h}\right)$$

$$z_{sh} = D_{sh}\left(f * G_{\sigma_0 sh}\right) = D_{sh}\left(f * G_{\sigma_0 h} * G_{\sigma_0 h\sqrt{s^2-1}}\right)$$

- Пусть s – целое. Тогда

$D_{sh}z = (D_h z)\downarrow_s$, где $\downarrow_s$ - оператор «прореживания»:

$$[z\downarrow_s]_{i,j} = z_{si,sj}$$

Повышение разрешения изображений

- Частный случай: задача понижения разрешения изображений в s раз: $h_1 = h$, $h_2 = hs$, $s \in \mathbb{N}$

$$z_h = D_h\left(f * G_{\sigma_0 h}\right)$$

$$z_{sh} = D_h\left(f * G_{\sigma_0 h} * G_{\sigma_0 h\sqrt{s^2-1}}\right)\downarrow_s$$

$$z_{sh} \approx D_h\left(f * G_{\sigma_0 h}\right) * G_{\sigma_0\sqrt{s^2-1}}\downarrow_s$$

$$z_{sh} = z_h * G_{\sigma_0\sqrt{s^2-1}}\downarrow_s$$

Повышение разрешения изображений

Повышение разрешения изображений

- Задача повышения разрешения изображений в s раз: $h_1 = h$, $h_2 = hs$, $s \in \mathbb{N}$

$$A_s z = u$$

где

$$A_s z = z * G_{\sigma_0 \sqrt{s^2 - 1}} \downarrow_s$$

- Метод регуляризации

$$z_\alpha = \arg\min_z \left(\|A_s z - u\| + \alpha \Omega[z] \right)$$

Повышение разрешения изображений

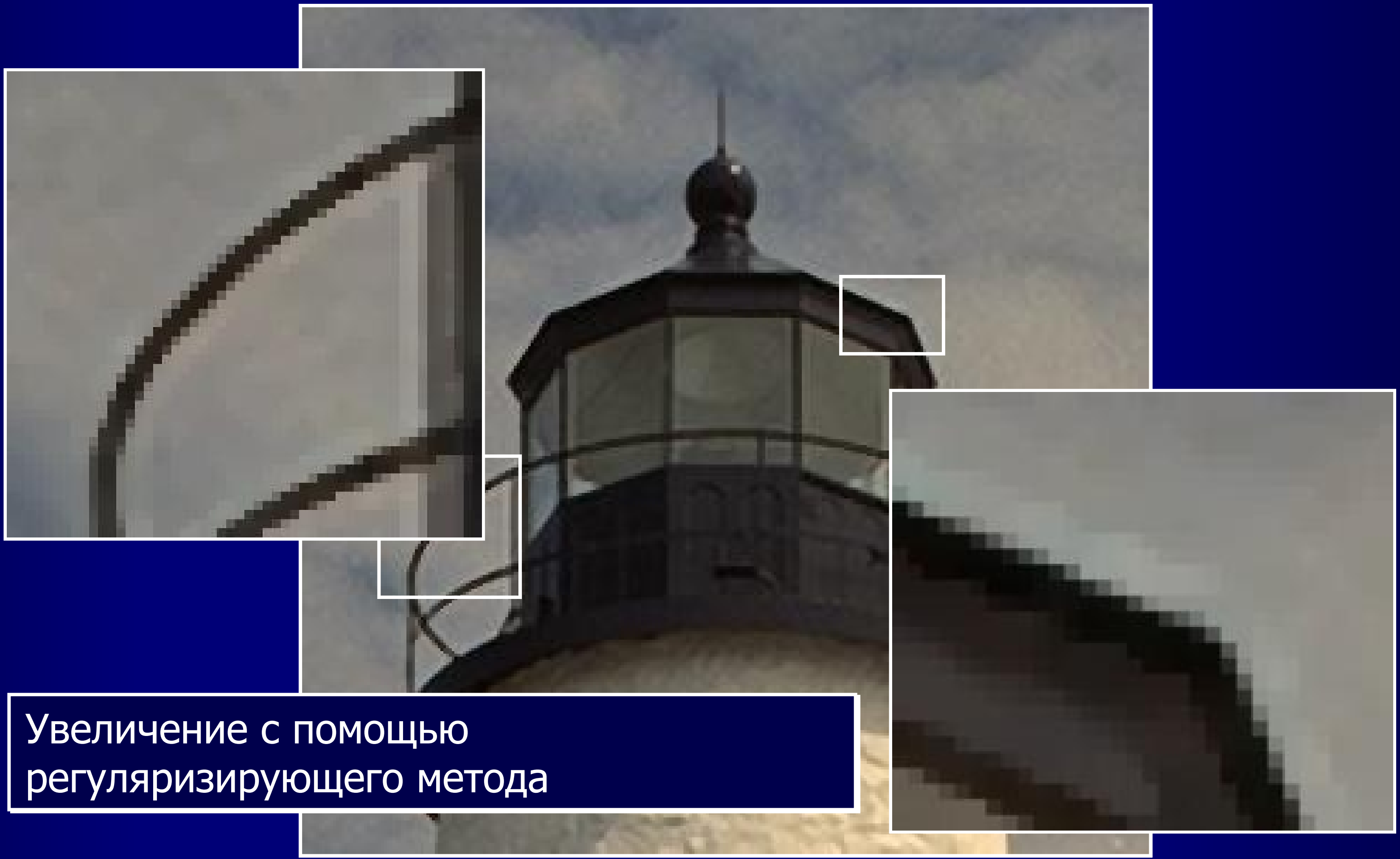
- В случае использования стабилизатора в виде полной вариации

$$\Omega[z] = \|\nabla z\|_1$$

- возможен явный выбор параметра регуляризации

$$z_R = \arg\min_z \|A_s z - u\| : \|\nabla z\|_1 \leq s\|\nabla u\|_1$$

Повышение разрешения изображений



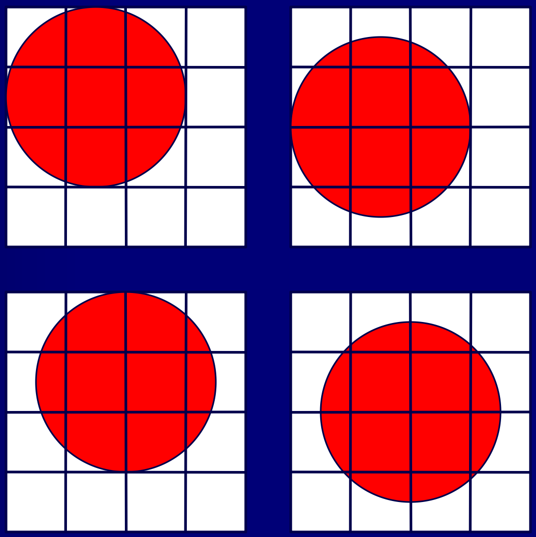
Увеличение с помощью регуляризирующего метода

Многокадровое суперразрешение

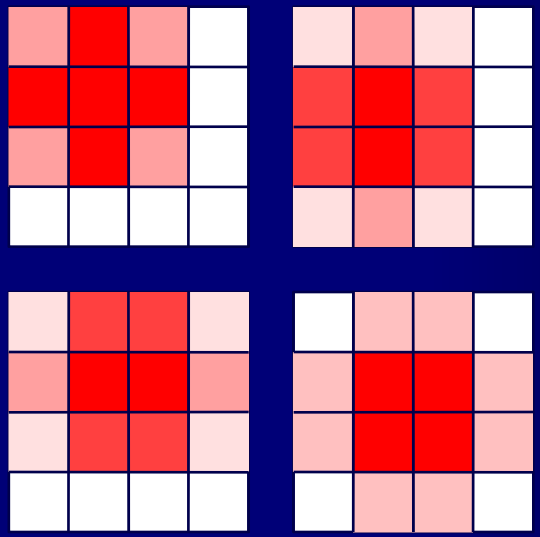
- Задача суперразрешения – это реконструкция изображения высокого разрешения по нескольким изображениям низкого разрешения

Многокадровое суперразрешение

- Источник дополнительной информации – субпиксельные сдвиги



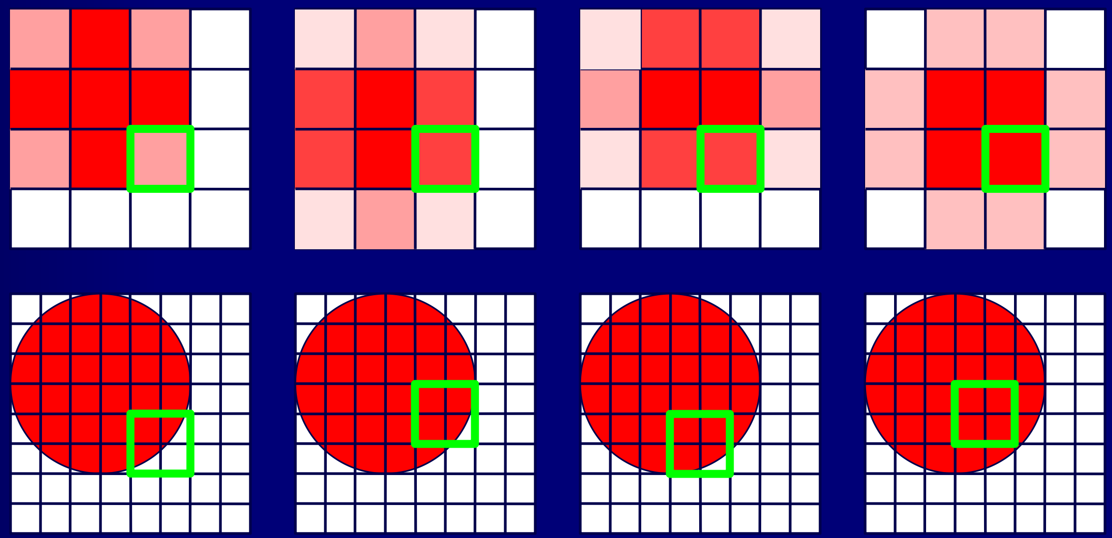
Движущийся объект



Получаемые с камеры изображения

Многокадровое суперразрешение

- Зная информацию о сдвигах, можно построить изображение более высокого разрешения

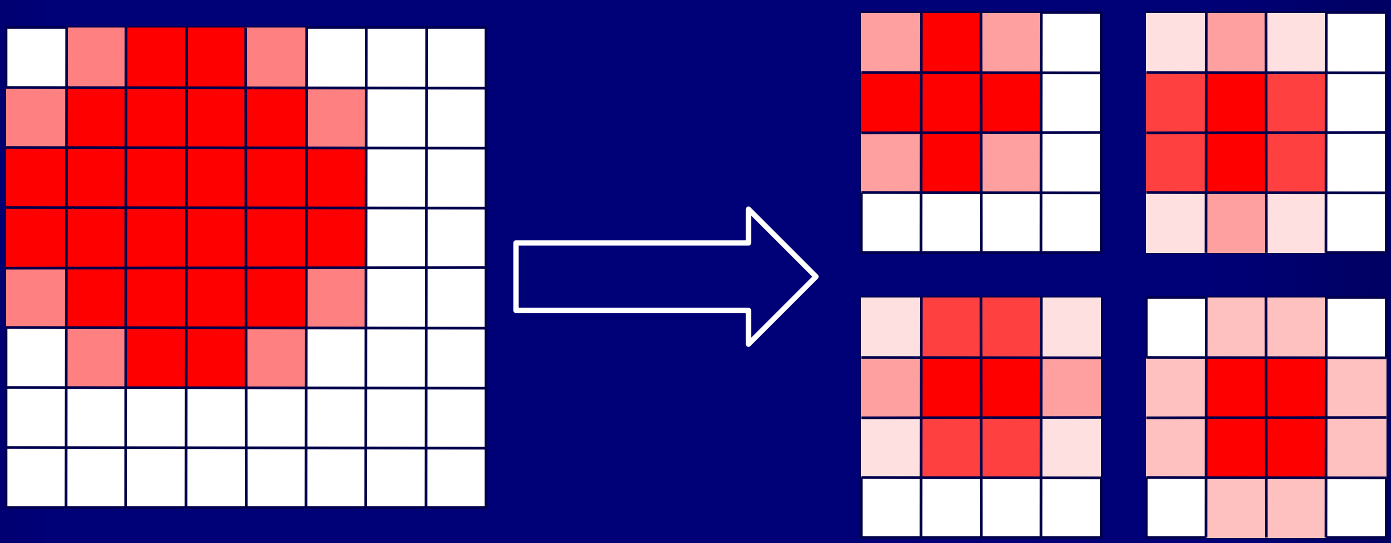


Получаемые с камеры изображения

Соответствие между пикселями изображений низкого разрешения и реконструируемого изображения высокого разрешения

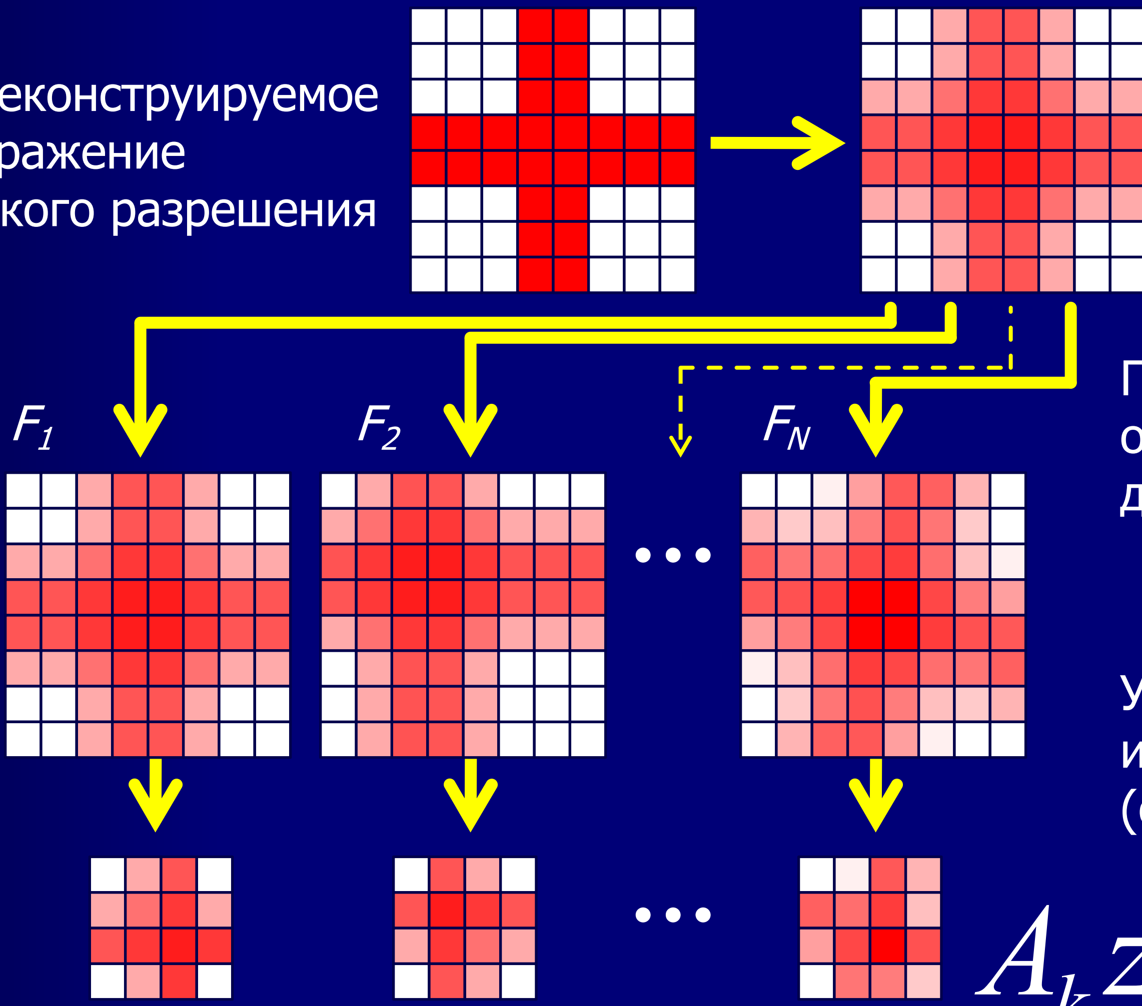
Многокадровое суперразрешение

- Построение через решение обратной задачи: найти такое изображение высокого разрешения, которое, будучи уменьшенным с учётом сдвигов, даст исходные изображения



Многокадровое суперразрешение

z – реконструируемое изображение высокого разрешения



Hz – размытое реконструируемое изображение

Применение операторов движения F_k

Уменьшение изображений (оператор D)

$$A_k z = D F_k H z$$

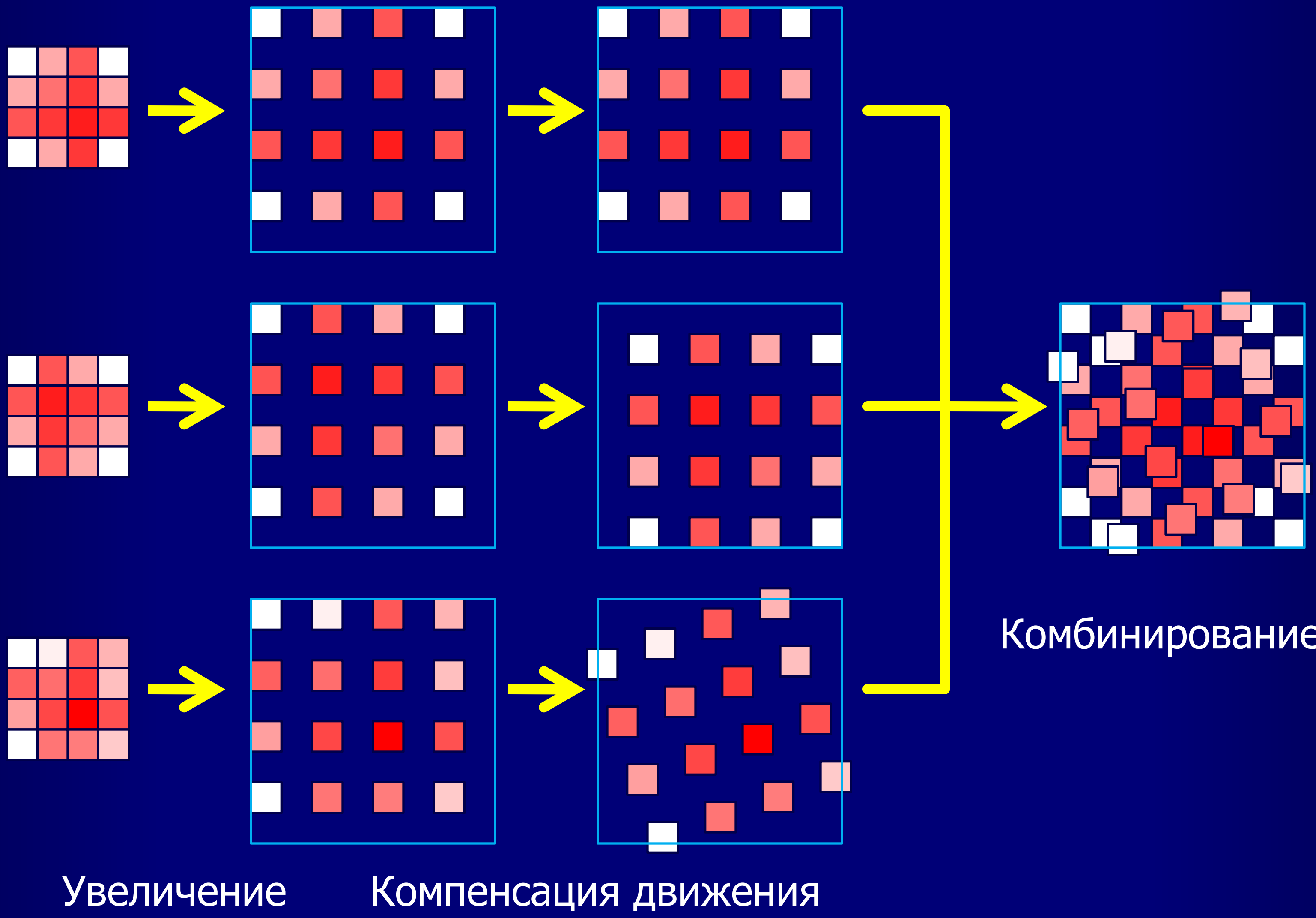
Многокадровое суперразрешение

- Регуляризирующий метод

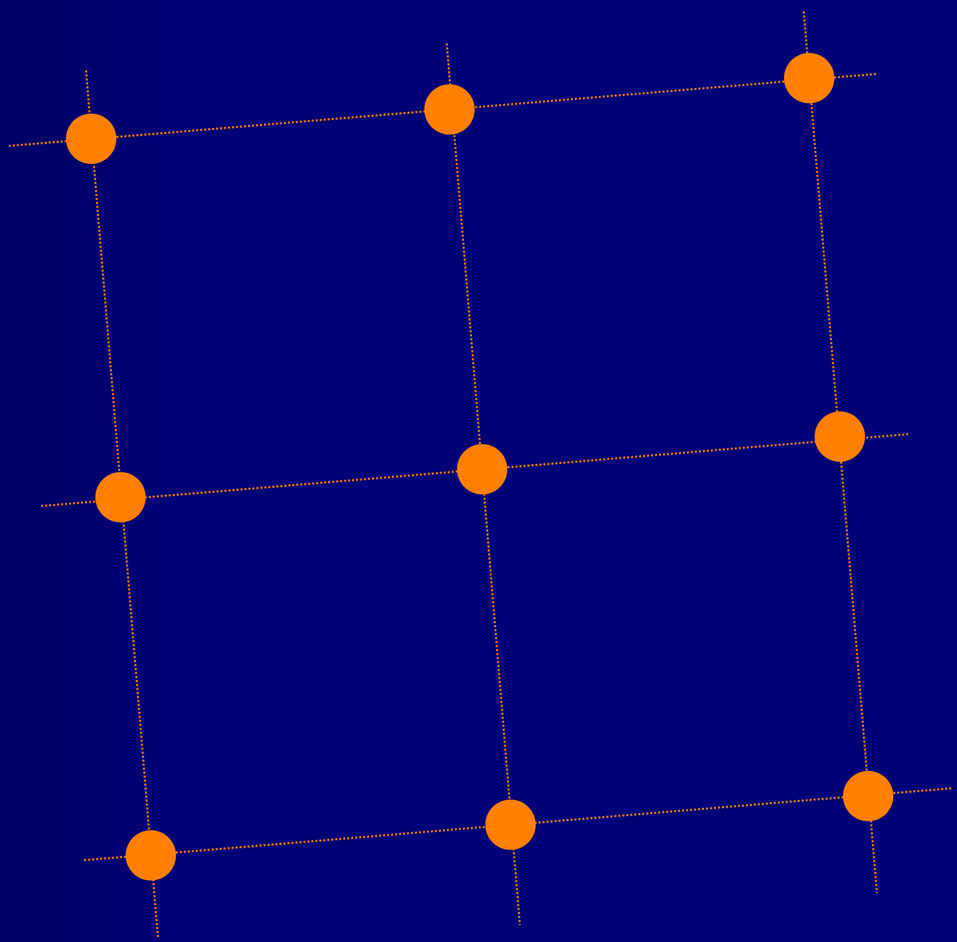
$$z_\alpha = \arg\min_z \left(\sum_k \|A_k z - u_k\| + \alpha\Omega[z] \right)$$

  – Несмотря на развитие алгоритмов, основанных на свёрточных нейронных сетях, данный метод сохраняет свою актуальность

Суперразрешение: Неитерационная реконструкция

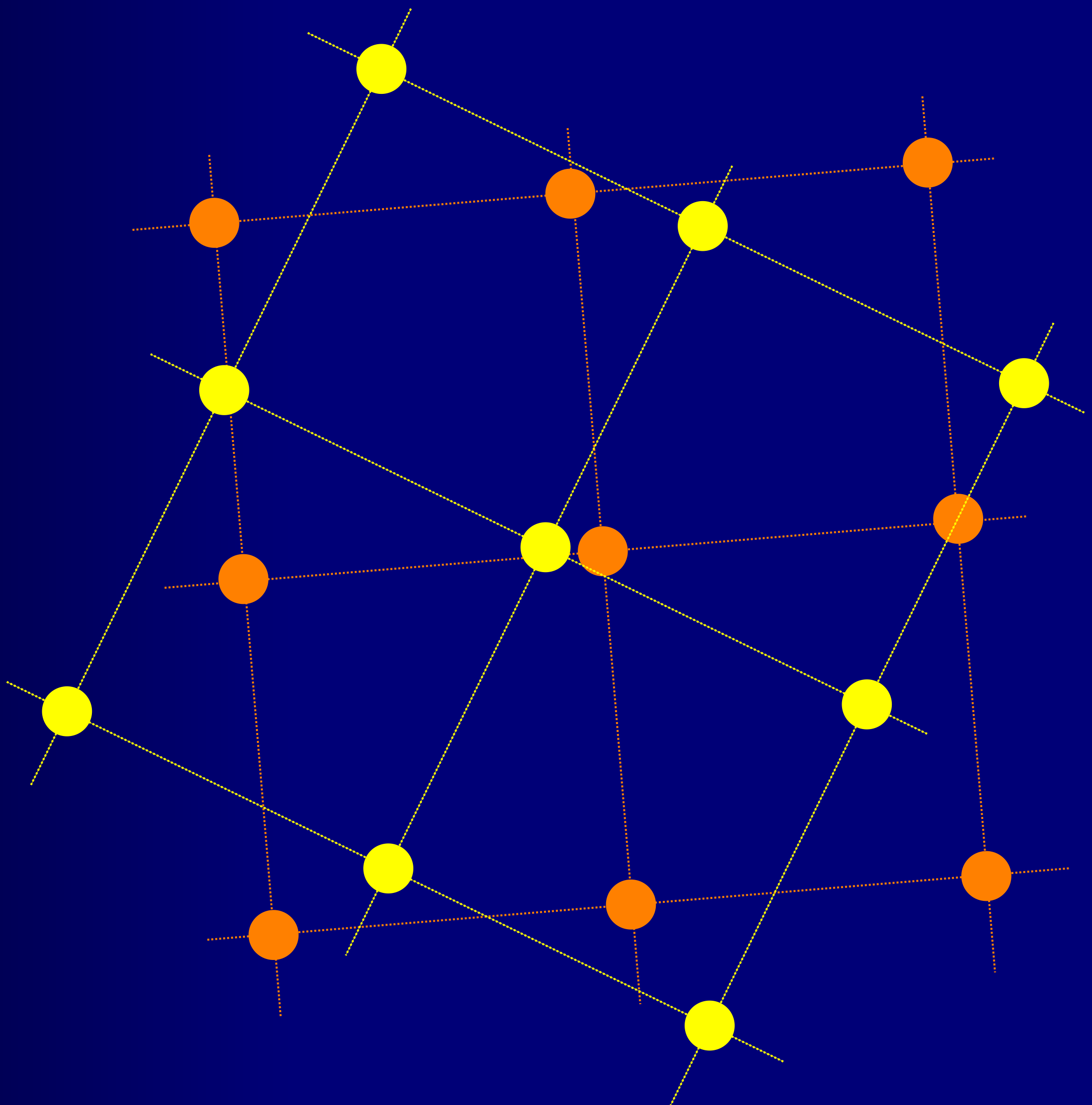


Суперразрешение: Неитерационная реконструкция



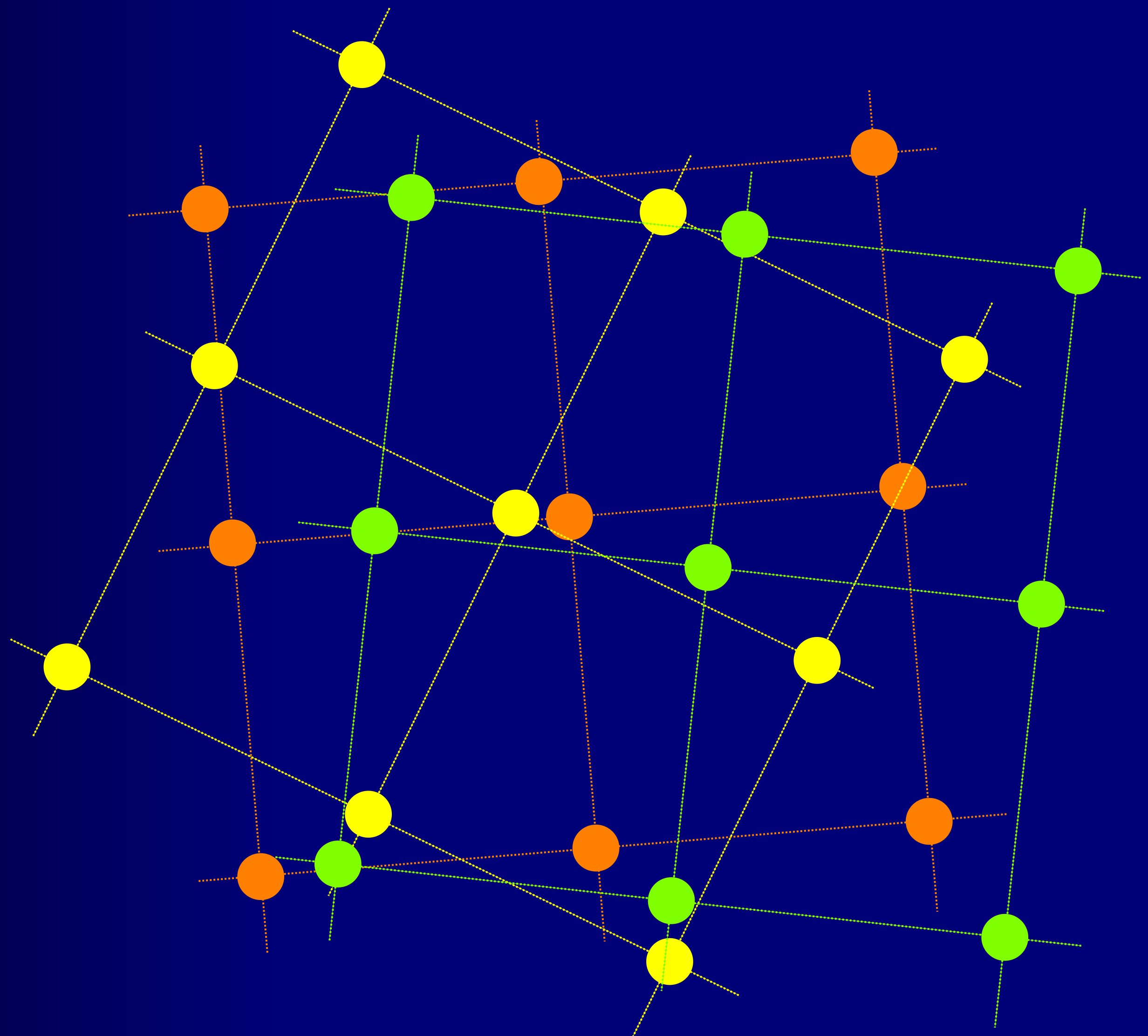
- Пиксели изображений низкого разрешения

Суперразрешение: Неитерационная реконструкция



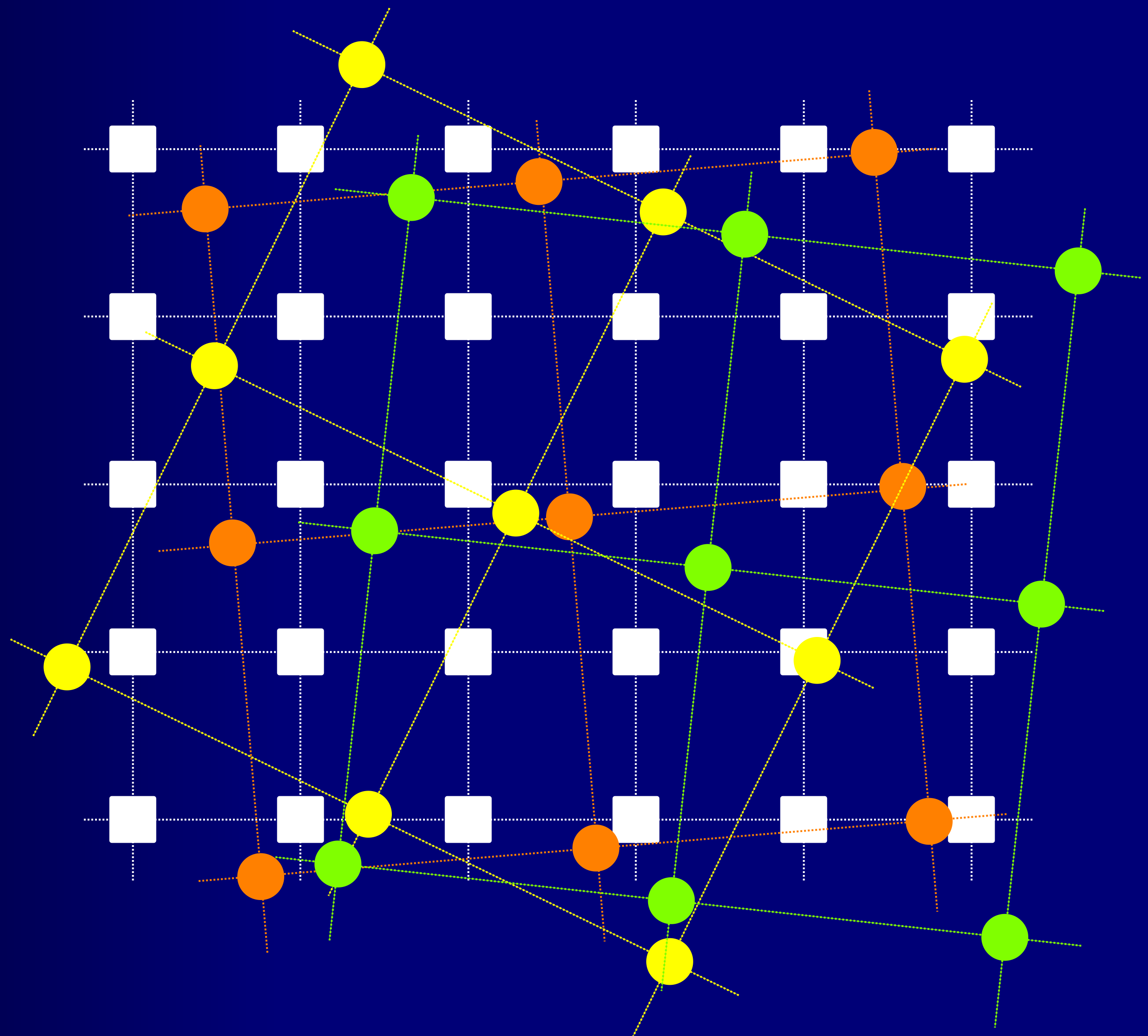
Пиксели изображений низкого разрешения

Суперразрешение: Неитерационная реконструкция



Пиксели изображений низкого разрешения

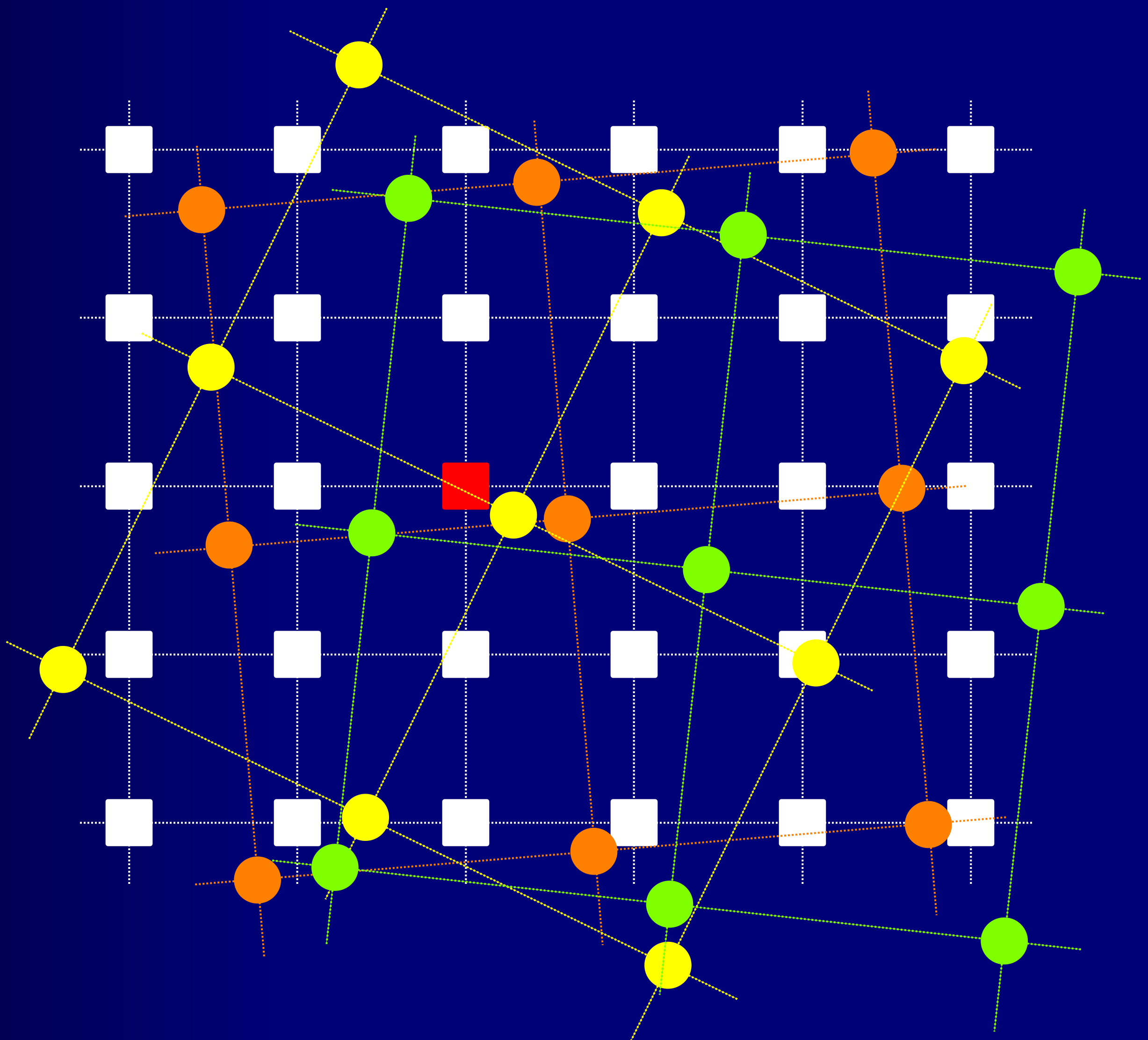
Суперразрешение: Неитерационная реконструкция



Пиксели (Hz)

Пиксели изображений низкого разрешения

Суперразрешение: Неитерационная реконструкция

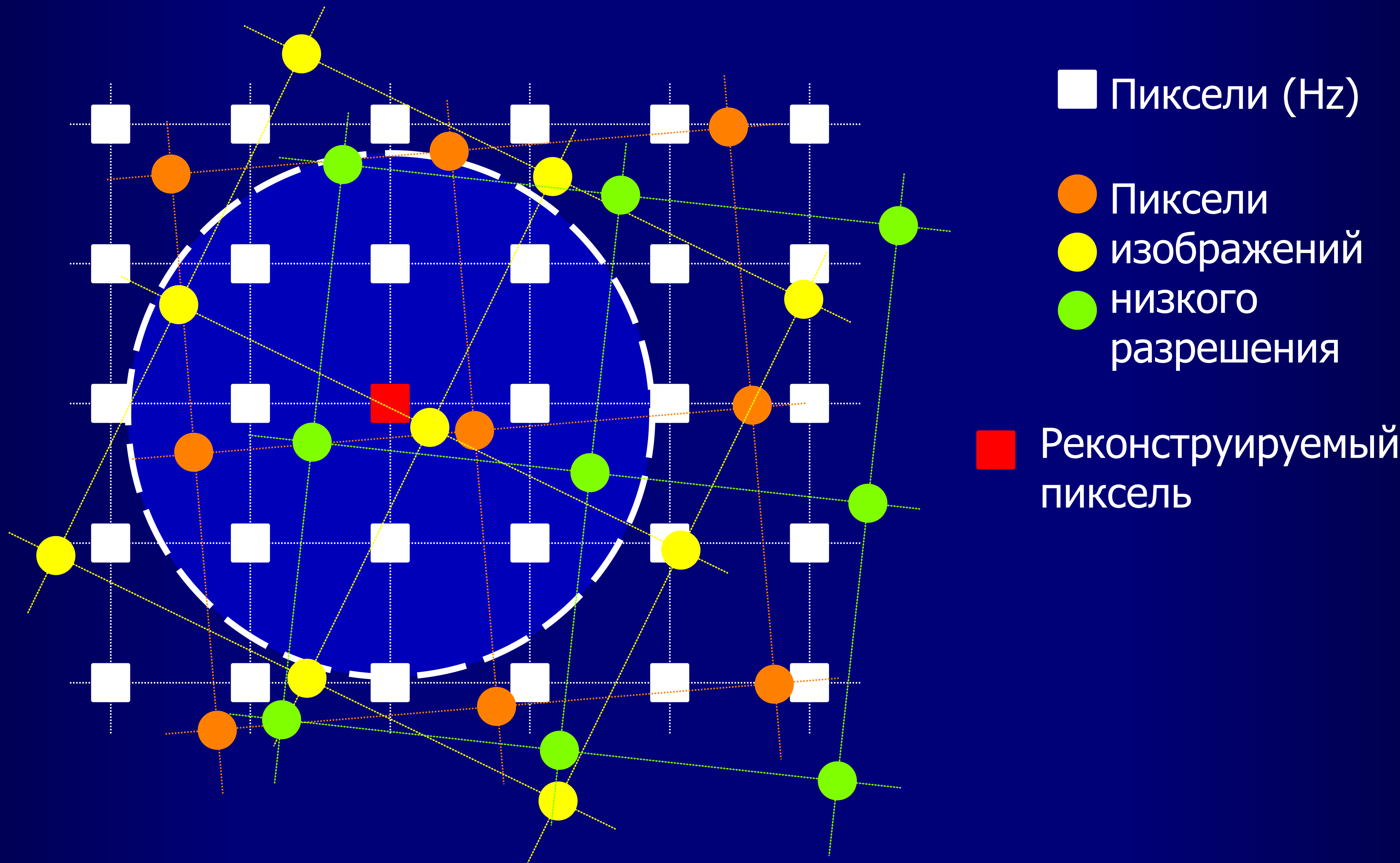


Пиксели (Hz)

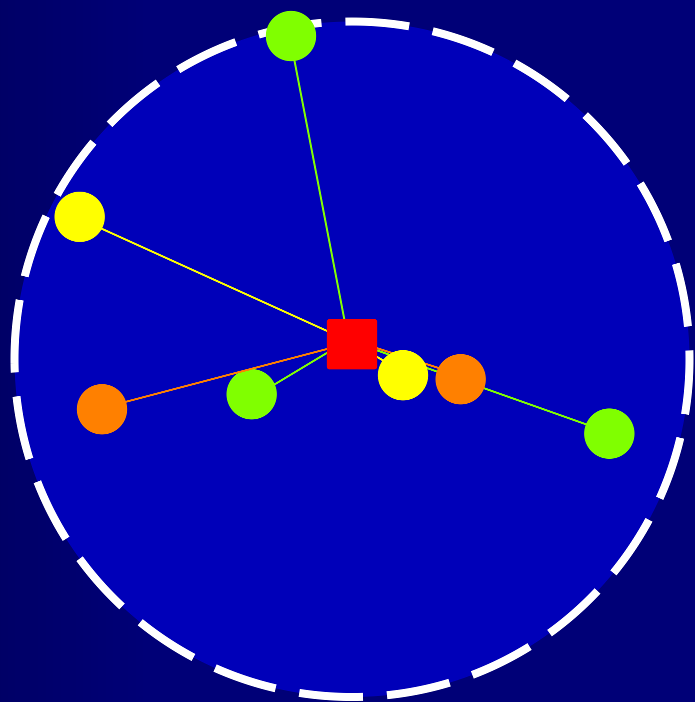
Пиксели изображений низкого разрешения

Реконструируемый пиксель

Суперразрешение: Неитерационная реконструкция



Суперразрешение: Неитерационная реконструкция

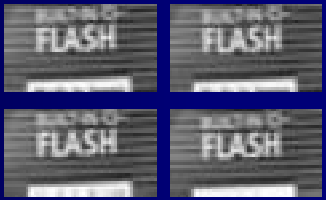


Пиксели изображений низкого разрешения

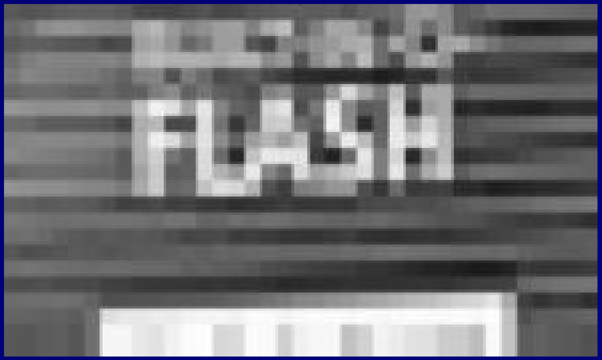
Реконструируемый пиксель

Для вычисления значения пикселя реконструируемого изображения используются попавшие в круг пиксели изображений низкого разрешения

Многокадровое суперразрешение



Исходные изображения



Метод ближайшего соседа

Регуляризирующий метод (1 изображение)

Приближённое решение (16 изображений)

Регуляризирующий метод (16 изображений)

Многокадровое суперразрешение

Интерполяция

Многокадровое суперразрешение

Повышение разрешения видео в 4 раза

Многокадровое суперразрешение

- Зависимость от числа кадров

Исходные кадры, увеличенные повторением пикселей

Результат суперразрешения

Нахождение векторов движения

- Основная проблема в многокадровом суперразрешении – нахождение оператора F_k

- Оптический поток – описание движение сцены с течением времени:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$

  – Перемещение пикселя с координатами (x, y) в момент времени t в пиксель с координатами $(x + \Delta x, y + \Delta y)$ в момент времени $t + \Delta t$

  – Δx, Δy – функции аргументов x, y, t.

Нахождение векторов движения

- Разложение в ряд Тейлора

$$I(x + \Delta\, x, y + \Delta y, t + \Delta t) =$$

$$= I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + o(\Delta x, \Delta y, \Delta t)$$

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0$$

  – Дифференцирование по Δt

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0$$

  – Требуются дополнительные условия для решения

Нахождение векторов движения

- Метод Лукаса-Канаде
  - Предположение: вектора движения постоянны в окрестности рассматриваемой точки

$$
\begin{gathered}
I_x(q_1)V_x + I_y(q_1)V_y = -I_t(q_1) \\
I_x(q_2)V_x + I_y(q_2)V_y = -I_t(q_2) \\
\dots \\
I_x(q_n)V_x + I_y(q_n)V_y = -I_t(q_n)
\end{gathered}
$$

Нахождение векторов движения

- Система линейных уравнений

$$AV = B$$

$$A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ I_x(q_2) & I_y(q_2) \\ \dots & \dots \\ I_x(q_n) & I_y(q_n) \end{bmatrix}$$

$$V = \begin{bmatrix} V_x \\ V_y \end{bmatrix}, \; B = \begin{bmatrix} -I_t(q_1) \\ -I_t(q_2) \\ \dots \\ -I_t(q_n) \end{bmatrix}$$

  – Система несовместна, используется метод наименьших квадратов

Нахождение векторов движения

- Метод наименьших квадратов

$$\|AV - B\|_2^2 \to \min$$

$$V = (A^T A)^{-1} A^T B$$

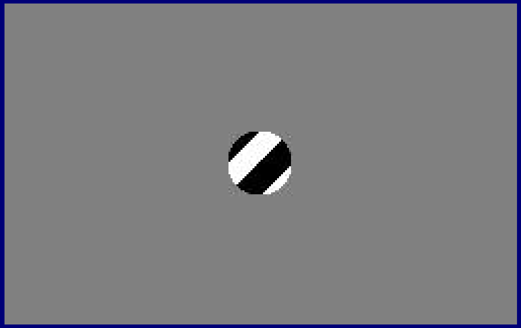
- Вариант с весовым окном
  - W – диагональная матрица $n \times n$, $W_{ii} = w_i$ – вес пикселя q_i

$$V = (A^T W A)^{-1} A^T W B$$

  - Пример весового окна – фильтр Гаусса

Нахождение векторов движения

- Ограничения метода Лукаса-Канаде
  - Только малые вектора смещения
    - Возможно использование иерархического подхода
  - Сложности со скачкообразным изменением векторов движения
  - Проблема апертуры



  - Обратимость матрицы, проверка собственных значений $\lambda_1 \geq \lambda_2 > 0$

Нахождение векторов движения

- Вариационный подход

$$V = [u(x, y),\ v(x,\ y)]^T$$

$$V_R = \arg\min_V \left(\iint \left(I_x u + I_y v + I_t\right)^2 + \alpha\Omega[u, v] \right)$$

- Проблема: минимизация
  - Невыпуклость, локальные минимумы

Нахождение векторов движения

- Метод Хорна-Шунка

$$\Omega[u,v] = \|\nabla u\|_2^2 + \|\nabla v\|_2^2$$

  - Уравнение Эйлера-Лагранжа

$$S(q) = \int_{\Omega} L\big(t, q(t),\ \dot{q}(t)\big)dt \to \min$$

$$q(t) = (q_1(t), q_2(t), \ldots, q_n(t))$$

$$\frac{\partial L}{\partial q_i}\big(t, q(t), \dot{q}(t)\big) - \frac{d}{dt}\frac{(\partial L)}{\partial q_i}\big(t, q(t), \dot{q}(t)\big) = 0$$

Нахождение векторов движения

- Метод Хорна-Шунка

$$I_x\left(I_x u + I_y v + I_t\right) - \alpha\Delta u = 0$$
$$I_y\left(I_x v + I_y v + I_t\right) - \alpha\Delta v = 0$$

$$\Delta u(x,y) \approx \overline{u}(x,y) - u(x,y)$$
$$\Delta v(x,y) \approx \overline{v}(x,y) - v(x,y)$$

$$(I_x^2 + \alpha)u + I_x I_y v = \alpha\overline{u} - I_x I_t$$
$$I_x I_y u + \left(I_y^2 + \alpha\right)v = \alpha\overline{v} - I_y I_t$$

Нахождение векторов движения

- Метод Хорна-Шунка – метод Якоби

$$u^{k+1} = \overline{u}^k - \frac{I_x\left(I_x\overline{u}^k + I_y\overline{v}^k + I_t\right)}{\alpha + I_x^2 + I_y^2}$$

$$v^{k+1} = \overline{v}^k - \frac{I_y\left(I_x\overline{u}^k + I_y\overline{v}^k + I_t\right)}{\alpha + I_x^2 + I_y^2}$$