

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования «Южно-Уральский государственный университет»
(национальный исследовательский университет)
ИОДО ЮУрГУ
Кафедра «ТТС»**

Презентация

**на тему: «Применение фильтров улучшения изображения в
автоматизированной системе видеонаблюдения»**

Анализ и разработка линейных фильтров для уменьшения шума на изображении

Методы фильтрации изображений (двухмерных сигналов) можно классифицировать следующим образом [1-3]:

1. Пространственная фильтрация:

линейная пространственная фильтрация (с различным размером маски фильтрации):

- метод скользящего среднего,
- метод взвешенного среднего.

нелинейная пространственная фильтрация (с различным размером маски фильтрации):

- медианная,
- метод первой производной,
- метод второй производной (лапласиан),
- эквализация,
- комбинированные фильтры.

адаптивная пространственная фильтрация:

- адаптивные локальные фильтры,
- адаптивные медианные фильтры,
- рекурсивные фильтры,
- адаптивные оптимальные фильтры.

2. Частотная фильтрация:

- низкочастотные фильтры,
- высокочастотные фильтры,
- режекторные (полосовые) фильтры.

3. Вейвлет фильтрация.

Анализ и разработка линейных фильтров для уменьшения шума на изображении

Линейная пространственная фильтрация основана на определении среднего значения в определенной окрестности, покрытой маской фильтра. Фильтрация исходного изображения проводится последовательно для каждого пикселя изображения, который накрывается маской фильтра, координаты центрального элемента которой совпадают с координатами фильтруемого пикселя изображения. В общем виде, отфильтрованное изображение $g(x,y)$ можно представить, как

$$g(x,y) = \frac{\sum_{s=-a}^a \sum_{t=-b}^b w(s,t) f(x+s,y+t)}{\sum_{s=-a}^a \sum_{t=-b}^b w(s,t)}$$

где $f(x+s,y+t)$ – исходное изображение, s и t – координаты маски фильтра, $w(s,t)$ – значения маски, $a=(m-1)/2$, $b=(n-1)/2$, где m и n размеры маски фильтра; минимальный размер маски $m \times n$ равен 3×3 пикселей.

Формула (1) описывает метод средне взвешенной фильтрации, а при $w(s,t)=1$ для всех координат маски – метод скользящего среднего.

Линейная пространственная фильтрация «убирает» высокие пространственные частоты, и поэтому уменьшает резкость изображения.

Нелинейная пространственная фильтрация основана на порядковых статистиках.

Медианную фильтрацию можно представить, как

$$g(x,y) = \text{med}_{(s,t) \in S_{xy}} \{f(x,y)\}$$

где S_{xy} – прямоугольная окрестность размерами $m \times n$ с центром в точке (x,y) .

Медианная фильтрация в меньшей степени «убирает» высокие пространственные частоты, и поэтому хорошо фильтрует артефакты и шумы с некоторым сохранением резкости изображения.

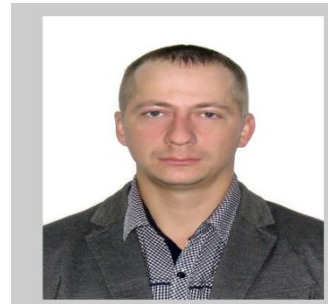
К фильтрам повышения резкости изображения относятся фильтры, основанные на первой и второй производной.

Анализ и разработка линейных фильтров для уменьшения шума на изображении

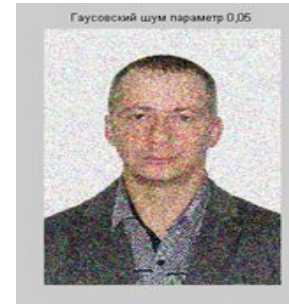
Особое место в практике фильтрации занимает фильтрация изображений с применением следующих фильтров:

1. Усредняющий фильтр 'average'.
2. Фильтр Гаусса 'gaussian'.
3. Фильтр Лапласа 'laplacian'.
4. Фильтр Лапласа-Гаусса 'log'.
5. Фильтр Собеля 'sobel'.
6. Фильтр Превита 'prewitt'.
7. Фильтр повышенной резкости 'unsharp'.
8. Медианный фильтр – medfilt2

Анализ и разработка линейных фильтров для уменьшения шума на изображении



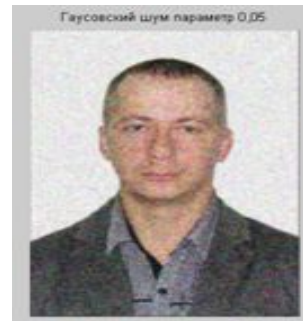
Исходное
изображение



Зашумленное
изображение



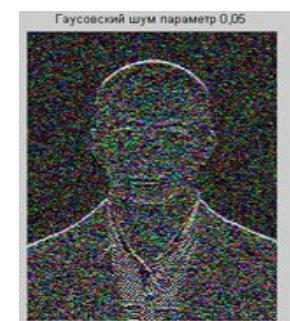
Фильтр 'average'
Маска 3×3 пикселей



Фильтр Гаусса 'gaussian'
Маска 3×3



Фильтр Лапласа 'laplacian'
Маска 3×3



Фильтр Собеля 'sobel'
Маска 3×3

Анализ и разработка линейных фильтров для уменьшения шума на изображении



Фильтр Превита 'prewitt'
Маска 3×3



Медианный фильтр
Маска 3×3

Анализ и разработка линейных фильтров для уменьшения шума на изображении

```
Листинг программы в MatLabclear all;
close all;
I2 =imread('1.gif');
I1 =imread('1.jpg');
imshow(I1);
%Белый шум
N1 = imnoise(I1, 'gaussian',0,0.05);
N1_2 = imnoise(I2, 'gaussian',0,0.05);
figure();
subplot(2,2,1);
imshow(N1);
title('Гаусовский шум параметр 0,05');
N2 = imnoise(I1, 'gaussian',0,0.01);
N2_2 = imnoise(I2, 'gaussian',0,0.01);
subplot(2,2,2);
imshow(N2);
title('Гаусовский шум параметр 0,01');
%Соль-перец
M1= imnoise(I1,'salt & pepper', 0.02);
M1_2= imnoise(I2,'salt & pepper', 0.02);
subplot(2,2,3);
imshow(M1);
title('Соль-перец параметр 0,02');
subplot(2,2,4);
M2= imnoise(I1,'salt & pepper', 0.04);
M2_2= imnoise(I2,'salt & pepper', 0.04);
imshow(M2);
title('Соль-перец параметр 0,04');
%-----
```

Анализ и разработка линейных фильтров для уменьшения шума на изображении

Выводы

1. Процедура фильтрации для всего изображения – это последовательное выполнение фильтрации каждой точки. То есть, окно фильтра "пробегает" по всем элементам фильтруемого изображения и в каждом своем положении формирует усредненное значение яркости по всем точкам, попавшим в окно.
2. Низкочастотные фильтры (НЧ) лучше справляются с белым шумом чем высокочастотные (ВЧ), потому что ВЧ фильтры не предназначены для борьбы с шумами. Разница в результатах обусловлена тем, что НЧ фильтр убирает шум, а ВЧ фильтр повышает резкость.
3. Медианная фильтрация подходит для импульсного шума потому что результат фильтрации уже не содержит отсчета, который явно выпадает из выборки, и, скорее всего, является помехой.
4. Применение фильтра prewitt к исходному изображению приводит к появлению выделения границ изображения в виде толстых светлых линий на черном фоне, фильтра average к исходному изображению приводит к небольшому размытию растровых точек, фильтра gaussian к исходному изображению приводит к появлению сильного размытия, которое приводит к ухудшению четкости и резкости, фильтра laplacian к исходному изображению приводит к появлению выделения границ изображения в виде светлых линий на черном фоне, фильтра sobel к исходному изображению приводит к появлению сильного выделения границ изображения в виде толстых светлых линий на черном фоне.
5. В рассмотренном случае зашумленности изображения наилучшим образом показал себя медианный фильтр.

Спасибо за внимание!