

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 13:12:44
Уникальный программный ключ:
65ab2aa0d384efe84806a4c088eddb8479e411a

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии



УТВЕРЖДАЮ:
проректор по учебной работе
Локтионова О.Г.
_____ 2022 г.

ВОССТАНОВЛЕНИЕ ИЗОБРАЖЕНИЙ ИЗОБРАЖЕНИЙ В
СРЕДЕ MATLAB

Методические указания для выполнения лабораторных работ по дисциплине «Методы и алгоритмы обработки изображений» для студентов направления подготовки 09.03.04 «Программная инженерия»

УДК 004.932

Составитель: Р.А. Томакова

Рецензент

Кандидат технических наук, доцент *А.В. Малышев*

Восстановление изображений в среде MATLAB:
методические указания для проведения лабораторных работ и выполнения самостоятельной внеаудиторной работы по дисциплине «Методы и алгоритмы обработки изображений» для студентов направления подготовки бакалавров 09.03.04 «Программная инженерия»/ Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2022. 18с.

Рассмотрена методика изучения моделирования искажений, вносимых линейной системой при формировании изображения и восстановление изображений на основе встроенных функций системы MATLAB.

Методические указания составлены в соответствии с федеральным государственным образовательным стандартом высшего образования направления подготовки 09.03.04 «Программная инженерия» и на основании учебного плана направления подготовки 09.03.04 «Программная инженерия» (профиль «Разработка программно-информационных систем»).

Предназначены для студентов, обучающихся по направлению подготовки 09.03.04 «Программная инженерия» (профиль «Разработка программно-информационных систем») всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать	Формат 60×84 1/16.		
Усл. печ. л.	. Уч. - изд. л.	. Тираж 25 экз. Заказ.	Бесплатно.
Юго-Западный государственный университет. 305040, г. Курск, ул. 50 лет Октября, 94.			

ВОССТАНОВЛЕНИЕ ИЗОБРАЖЕНИЙ

Цель и задачи лабораторного занятия (лабораторной работы): изучение функций среды MATLAB, применяемых для моделирование искажений, вносимых системой при формировании изображений;

освоение процедур восстановления изображений;
приобретение практических умений и навыков их использования.

Планируемые результаты обучения (формируемые знания, умения, навыки и компетенции):

Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной :
ПК-4.4 Выполняет тестирование программного обеспечения

Необходимые материально-техническое оборудование и материалы:

1. Класс ПЭВМ - Athlon 64 X2-2.4; Cel 2.4, Cel 2.6, Cel 800.
2. Мультимедиа центр: ноутбук ASUS X50VL PMD T2330/14"/1024Mb/ 160Gb/ сум-ка/проектор inFocus IN24+ .
3. Экран мобильный Draper Diplomat 60x60
4. Доступ в сеть Интернет.

Шкала оценивания и критерии оценивания выполненной лабораторной работы:

Форма контроля	Минимальный балл		Максимальный балл	
	балл	примечание	балл	примечание
Лабораторная работа № 1 (Работа с файлами различных типов изображений в среде MATLAB)	2	Выполнил, но «не защитил»	4	Выполнил и «защитил»

План проведения лабораторного занятия (лабораторной работы)

1. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ

1. Восстановить «размытое» изображение файла `construc.bmp`, обусловленное турбулентностью атмосферы, гомоморфным фильтром.

2. Создать М-файл для демонстрации восстановления искажений изображения, обусловленных турбулентностью атмосферы и восстановить его фильтром Винера, инверсным или гомоморфным фильтром с возможностью задания файла изображения, коэффициента нерезкости и отношения сигнал/шум с клавиатуры.

3. Добавьте к исходному выражению шум типа «соль», «перец». Примените зашумленному изображению медианный фильтр. Сравните результаты медианной и усредняющей фильтрации для помехи данного вида.

4. Выполнить моделирование размытого зашумленного изображения, используя приведенный алгоритм.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1 Модель процесса искажения/восстановления изображения

Одной из задач обработки изображений является задача устранения искажений, возникающих в процессе их формирования. В процессе записи изображения искажаются также шумами, присутствующими в любом реальном физическом устройстве [1,5,7,8].

Модель процесса искажения представлена на рисунке 1, она предполагает действие некоторого искажающего оператора H на исходное изображение $f(x, y)$ и после добавления аддитивного шума получают искаженное изображение $g(x, y)$.

Задача восстановления заключается в построении некоторого приближения $\hat{f}(x, y)$ исходного изображения по заданному (искаженному) изображению $g(x, y)$ при наличии искажающего оператора H и аддитивного шума $\eta(x, y)$.

Требуется, построить такое приближение, чтобы оно было как можно ближе к исходному изображению.

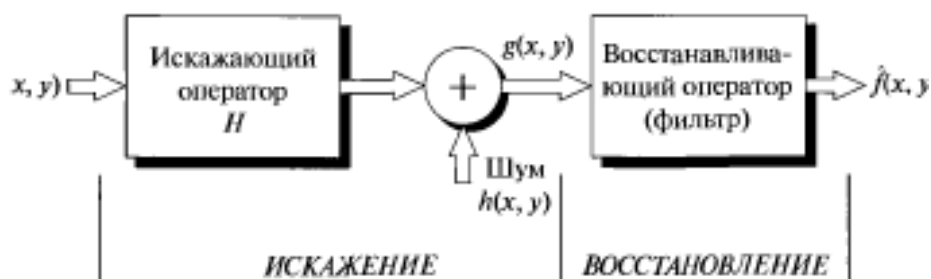


Рисунок 2.1. Модель процесса искажения/восстановления изображения

Искаженное изображение может быть представлено в пространственной области в виде

$$g(x, y) = f(x, y) \otimes h(x, y) + \eta(x, y) \quad , \quad (1)$$

где $g(x, y)$ - искаженное изображение; $h(x, y)$ - функция, представляющая искажающий оператор в пространственной области; $f(x, y)$ - исходное изображение; $\eta(x, y)$ - аддитивный шум.

Таким образом, **выходное изображение $g(x, y)$ представляет собой свертку входного изображения с импульсным откликом системы, искаженную аддитивным шумом $\eta(x, y)$.**

В частотной области модель имеет вид:

$$G(u, v) = H(u, v)F(u, v) + N(u, v), \quad (2)$$

где заглавными буквами обозначены Фурье образы соответствующих функций.

Фильтр Винера

Фильтр Винера представляет собой один из первых методов, разработанных для сокращения уровня случайных шумов на изображениях. При этом предполагается, что аддитивный шум — это стационарный случайный процесс, независимый от расположения пикселя. Алгоритм фильтрации минимизирует квадратичную ошибку между исходным и восстановленным изображением. Фильтрация Винера является фильтром низких частот, но вместо использования единственного отсечения частоты фильтр использует нижнюю частоту среза в областях с малой детализацией и верхнюю частоту среза для сохранения деталей в областях с кромками или другими элементами с высокими отклонениями. Размер области определяет

общую частоту среза: большее окно соответствует нижним частотам среза, а значит, большему подавлению размытости и шума.

В данном алгоритме пиксель y в отфильтрованном изображении наследуется от пикселя x во входном зашумленном изображении с помощью следующего преобразования:

$$y = \mu_x + (x - \mu_x) \frac{v_x}{v_x + v_n}$$

где μ_x и v_x — среднее значение и дисперсия x в окрестностях пикселя, а v_n — дисперсия аддитивного шума, полученного из входного изображения.

Каждый выходной пиксель является суммой локального среднего значения из окрестности входного пикселя и локального значения контрастности $(x - \mu_x)$, отмасштабированного так, что в областях с большой детализацией, где дисперсия шума (v_n) намного меньше, чем дисперсия изображения (v_x), коэффициент масштабирования очень близок к 1, а выходной пиксель y очень близок к входному пикселю x с небольшой фильтрацией, в то время как в областях с низкой детализацией, где дисперсия изображения ниже, выходной пиксель больше похож на локальное среднее значение (т. е. фильтрация низких частот).

Границы изображения считаются нулевыми значениями в оттенках серого, что делает выходные пиксели рядом с границами изображения недействительными (до размера области окрестности). Также область окрестности не должна быть больше входного изображения.

Задача построения восстанавливающей системы состоит в поиске импульсного отклика $r(x, y)$, минимизирующего средний квадрат ошибки J :

$$J = [f(x, y) - g(x, y) \cdot r(x, y)]^2. \quad (3)$$

Исходя из минимизации среднего квадрата ошибки J и рассматривая исходное изображение $f(x, y)$ и результаты его преобразований как действительные стационарные случайные сигналы, импульсный отклик восстанавливающей системы в частотной области определяется из следующего выражения:

$$W_g(u, v) \cdot R(u, v) = W_{gf}(u, v), \quad (4)$$

где $W_g(u, v)$ – спектр мощности случайного процесса; $R(u, v)$ – частотная функция восстанавливающей системы; $W_{gf}(u, v)$ – кросс-спектр мощности процессов g и f .

Тогда частотная функция восстанавливающей системы определяется как

$$R(u, v) = \frac{W_{gf}(u, v)}{W_g(u, v)}. \quad (5)$$

После учета шумов частотная функция принимает вид:

$$R(u, v) = \frac{H^*(u, v)}{|H(u, v)|^2 + \frac{W_n(u, v)}{W_f(u, v)}}, \quad (6)$$

где $H(u, v)$ – частотная функция линейной искажающей системы; $(...)^*$ – символ комплексного сопряжения; $W_f(u, v)$ – спектр мощности исходного изображения; $W_n(u, v)$ – спектр мощности шума.

Система с такой частотной функцией известна как **фильтр Винера**. Такая же система с соотношением спектров мощности исходного изображения и шума (шум/сигнал), равным нулю, называется *инверсным фильтром*.

При съемке космических объектов через атмосферу Земли происходит “размытие” изображения, обусловленное турбулентностью атмосферы. Этот эффект приближенно можно представить действием линейной системы с импульсным откликом вида:

$$h(x, y) = \exp\left(-\frac{x^2 + y^2}{2\rho^2}\right), \quad (7)$$

которому соответствует частотная характеристика

$$H(u, v) = \exp(-2\rho^2(u^2 + v^2)), \quad (8)$$

где ρ – коэффициент нерезкости.

Задание 1. Смоделировать «размытие» изображения, обусловленное турбулентностью атмосферы, и восстановить его фильтром Винера:

```
[X,map] = imread('c:\Image\Athena.bmp');
I = im2double(ind2gray(X,map));
```

```

figure,imshow(I)
title('original image')
[f11,f22] = freqspace([15 15],'meshgrid');
a = 1;
H = exp(-a^2.*(f11.^2+f22.^2));
h = fsamp2(f11,f22,H,[5,5]);
Id = conv2(I,h,'same');
figure,imshow(Id)
title('defect image')
k = 1e-4;
[HT,f1,f2] = freqz2(h,[5 5]);
HV = conj(HT)./(abs(HT).^2+k);
hv = fsamp2(HV);
Ir = conv2(Id,hv,'same');
figure,imshow(Ir)
title('restore image')

```

В пакете IPT функция, выполняющая *винеровскую фильтрацию*, называется **deconvwnr**.

Синтаксис

Ir = deconvwnr(Id, h, k),

где **Id** – искаженное изображение; **h** – импульсный отклик линейной искажающей системы; **k**– параметр, который определяет отношение спектра мощности шума и исходного изображения. По умолчанию соотношение шум/сигнал **k = 0** (т. е. форма винеровского фильтра совпадает с инверсным фильтром).

Для выполнения восстановления изображения файла Athena.bmp из предыдущего задания форма вызова функции совпадает с приведенным синтаксисом:

```
>> Ir = deconvwnr(Id, h, k);
```

Гомоморфный фильтр

Для восстановления изображения также можно использовать гомоморфный фильтр.

При проектировании фильтра отыскивается такая линейная оценка

$$\hat{f}(x, y) = L[g(x, y)],$$

(где L – линейный оператор), чтобы энергетический спектр оценки равнялся энергетическому спектру исходного изображения. Пространственно- частотная характеристика гомоморфного фильтра определяется следующим выражением:

$$H_p(u, v) = \left(\frac{1}{|H(u, v)|^2 + \frac{W_n(u, v)}{W_f(u, v)}} \right)^{1/2}$$

Задание 2.

2.1. Изображения, сформированные различными оптико-электронными системами, во многих случаях искажаются действием различных помех. *Ослабление действия помех осуществляется с помощью фильтров.* Рассмотрите пример использования так называемого *усредняющего фильтра Гаусса.*

Для этого наложите на исходное изображение *гауссовский шум* командой

J2 = imnoise(I,'gaussian',0,0.05);

где среднее значение шума равно **0**, а дисперсия - **0,05**.

Просмотрите полученное изображение. Затем командой

K = filter2(fspecial('average',3),J2)/1;

примените усредняющий фильтр, который выполняет усреднение пикселей в квадрате 3 на 3 элемента с рабочим элементом в центре. Сравните полученное изображение с предыдущим.

Используя команду

K = filter2(fspecial('average',5),J2)/4;

примените усредняющий фильтр, который выполняет усреднение пикселей в квадрате 5 на 5 элементов с рабочим элементом в центре. Сравните полученное изображение с предыдущим.

2.2. Примените к зашумленному гауссовским шумом изображению *адаптивный винеровский фильтр* командой

K1 = wiener2(J2,[5 5]);

Такой фильтр осуществляет адаптивную локальную обработку изображения в квадрате 5 на 5 элементов. Если значение среднеквадратического отклонения интенсивностей пикселей в данной локальной области большое, то `wiener2` выполняет небольшое

сглаживание. Если же это отклонение небольшое, то сглаживание больше. Этот фильтр часто бывает более эффективным, чем обычная линейная фильтрация. Выведите изображение после адаптивной обработки и сопоставьте результат с действием усредняющего фильтра.

2.3. Добавьте к исходному выражению командой
M= imnoise(I,' salt & pepper', 0.02);

шум типа «salt & pepper (соль и перец)». Просмотрите изображение. Снова примените усредняющий фильтр, как и в предыдущем пункте. Сделайте соответствующий вывод по результатам данной обработки.

3. Примените теперь к предыдущему зашумленному изображению так называемый *медианный фильтр* командой :

L = medfilt2(M,[3 3]); .

Сравните результаты медианной и усредняющей фильтрации для помехи данного вида.

4. Загрузите командой **I = imread('saturn.tif');** и отобразите изображение из файла 'saturn.tif' .

Для повышения резкости изображения командами

H = fspecial('unsharp');
sharpened = imfilter(I,H);
imshow(sharpened);
title('Результат обработки');

Выполните фильтрацию исходного изображения и отображение результата. Прокомментируйте итоги.

Размывание изображений можно смоделировать с помощью функций **fspecial** из пакета IPT:

PSF= fspecial ('motion', len, theta).

Эта форма вызова **fspecial** возвращает PSF, которая аппроксимирует эффекты линейного перемещения камеры на *len* пикселей. Угловой параметр *theta* измеряется в градусах, отсчитывается против часовой стрелки.

Значения по умолчанию: **len=9** и **theta=0**, что соответствует смещению на 9 пикселей в горизонтальном направлении.

Для построения изображения, искаженного фильтром PSF используем команду:

>> g=imfilter (f, PSF, 'circular');

Параметр **'circular'** используется для подавления граничных эффектов. После чего искажение вносится в изображение в виде аддитивного шума по формуле:

```
>> g = g + noise;
```

где **noise** – случайное шумовое изображение, имеет тот же размер, что и **g**.

5. Моделирование размытого зашумленного изображения

1. Построить изображение с помощью команды:

```
>> f = checkerboard (8);
```

2. Получить размытое изображение командами:

```
>> PSF = fspecial ('motion', 12, 30);
```

```
>> gb = imfilter (f, PSF, 'circular');
```

3. Получить шумовое изображение, сгенерированное командой:

```
>> noise = imnoise (zeros (size (f)), 'gaussian', 0, 0.001 );
```

4. Смазанное и зашумленное изображение строится по команде:

```
>> gb =im2uint8( g + noise);
```

6. Восстановить размытое зашумленное изображение с помощью команды:

```
>>fr1=deconvwnr (g, PSF);
```

где **g** –это искаженное изображение, а **PSF** – функция разброса точек, **fr1**- это результат прямого применения инверсной фильтрации.

7. Получить число **R** по формулам

```
>>Sn=abs (fft2(noise)).^2;
```

```
>>nA=sum (Sn(:))/prod(size(noise));
```

```
>>Sf= abs (fft2(f)).^2;
```

```
>> fA=sum (Sn(:))/prod(size(noise));
```

```
>>R=nA/fA;
```

Для восстановления изображения с помощью этого соотношения **R** выполнить команду:

```
>>fr2=deconvwnr (g, PSF, R);
```

8. Использовать автокорреляционные функции при восстановлении изображения:

центрированная автокорреляционная функция шума

```
>> NCORR=fftshift (real (ifft2 (Sn)));
```

центрированная автокорреляционная функция изображения

```
>> ICORR=fftshift (real (ifft2 (Sf)));
```

```
>> fr3=deconvnr (g, PSF, NCORR, ICORR);
```

Сделать выводы по работе.

3. ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАДАНИЙ

1. Восстановить «размытое» изображение файла, обусловленное турбулентностью атмосферы, гомоморфным фильтром.

```
>> A = imread('C:\Users\Oracle\Desktop\обработка
изображения\лаб7\k','jpg');
[X, map] = rgb2ind(A,256);
I = im2double(ind2gray(X,map));
[f11,f22] = freqspace([15 15],'meshgrid');
a = 1;
H = exp(-a^2.*(f11.^2+f22.^2));
h = fsamp2(f11,f22,H,[30 30]);
Id = conv2(I,h,'same');
k = 1e-4;
[HT,f1,f2] = freqz2(h,[30 30]);
HP = sqrt(1./(abs(HT).^2+k));
hv = fsamp2(HP);
Ir = conv2(Id,hv,'same');
subplot(1,3,1),subimage(I), title('Исходноеизображение');
subplot(1,3,2),subimage(Id), title('Размытоеизображение');
subplot(1,3,3),subimage(Ir), title('Восстановленное изображение')
```



Рисунок 3.1- – Исходное, размытое и восстановленное изображения

2. Создать М-файл для демонстрации восстановления искажений изображения, обусловленных турбулентностью атмосферы и восстановить его фильтром Винера, инверсным или гомоморфным фильтром с возможностью задания файла изображения, коэффициента нерезкости и отношения сигнал/шум с клавиатуры.

```

function Vin(File_name,LEN, THETA, estimated_nsr)
A=imread(File_name);
I = im2double(rgb2gray(A));
% Simulate a motion blur.
% LEN = 21;
% THETA = 11;
PSF = fspecial('motion', LEN, THETA);
Id = imfilter(I, PSF, 'conv', 'circular');
Ir = deconvwnr(Id, PSF, estimated_nsr);
subplot(1,3,1),subimage(I), title('Original image');
subplot(1,3,2),subimage(Id), title('Defected image');
subplot(1,3,3),subimage(Ir), title('Restore image')
end
>>
Vin('C:\Users\Oracle\Desktop\обработкаизображения\лаб7\k.jpg',21,1
1,0);

```

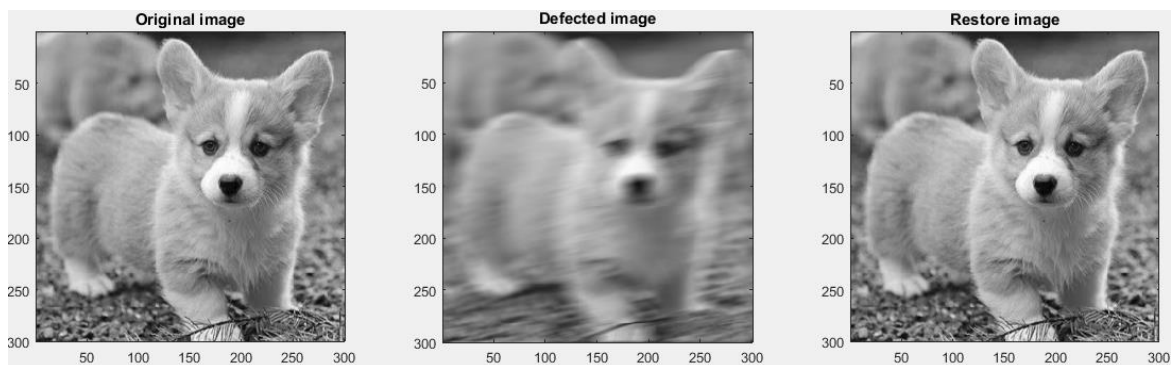


Рисунок 3.3 Исходное, размытое и восстановленное изображения

```

Vin('C:\Users\Oracle\Desktop\обработкаизображения\лаб7\k.jpg',30,21,
0.01);

```

3. Добавьте к исходному выражению шум типа «соль», «перец». Примените зашумленному изображению медианный фильтр. Сравните результаты медианной и усредняющей фильтрации для помехи данного вида.



Рисунок 3.4– Зашумленное изображение, отфильтрованное медианным фильтром и усредняющим фильтром

3. Выполнить моделирование размытого зашумленного изображения, используя приведенный алгоритм.

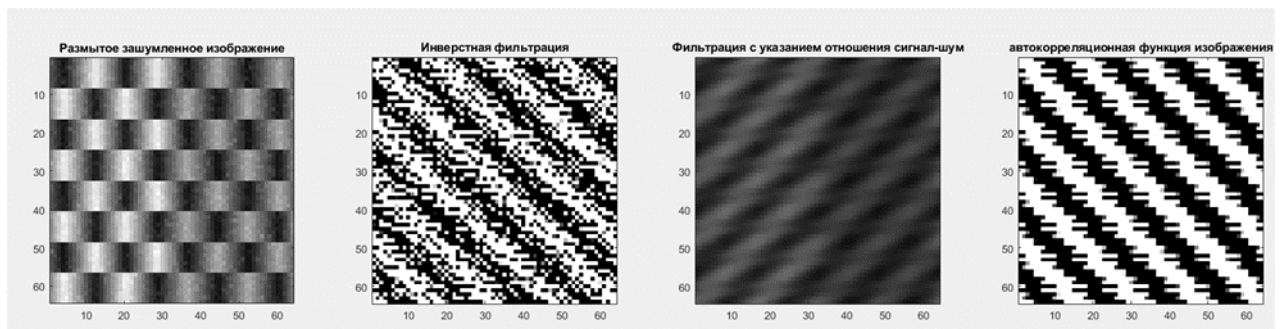


Рисунок 3.5 - Моделирование размытого зашумленного изображения

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что обуславливает искажения изображения при его формировании?
2. В чем заключается задача построения восстанавливающей системы?
3. Какие функции среды MATLAB используются для моделирования «размытия» изображения, обусловленное турбулентностью атмосферы?
4. Какой фильтр называется инверсным?
5. Сформулировать критерии восстанавливающей системы.
6. Каким образом может быть представлено искаженное изображение?
7. Записать вид модели процесса искажения/восстановления изображения.
8. В чем заключается задача восстановления изображения?
9. Какая функция среды MATLAB используются для осуществления винеровской фильтрации изображения?
10. Какая функция среды MATLAB осуществляет зашумление изображения типа «соль», «перец»?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Томакова, Р. А. Методы и алгоритмы цифровой обработки изображений : учебное пособие для студентов всех форм обучения по направлениям подготовки 09.03.03 Прикладная информатика; 09.03.04 «Программная инженерия», 09.04.04 «Программная инженерия» (профиль «Разработка информационно-вычислительных систем») / Р. А. Томакова, Е. А. Петрик ; Юго-Зап. гос. ун-т. - Курск : Университетская книга, 2020. - 310 с. - Загл. с титул. экрана. – Текст : электронный.

3. Яне, Бернд. Цифровая обработка изображений : [учебное пособие] / пер. с англ. А. М. Измайловой. - М. : Техносфера, 2007. - 584 с. : ил. + 1 эл. опт. диск (CD-ROM). - (Мир цифровой обработки. XI. 06). - Библиогр.: с. 575-583 (221 назв.). - ISBN 978-5-94836-1 22-2 : 285.00 р. - Текст : непосредственный.

4. Красильников, Н. Н. Цифровая обработка 2D- и 3D-изображений : учебное пособие / Н. Н. Красильников. - СПб. : БХВ-Петербург, 2011. - 608 с. : ил. - ISBN 978-5-9775-07 00-4 : 372.50 р. - Текст : непосредственный.

5. Томакова, Римма Александровна. Интеллектуальные технологии сегментации и классификации биомедицинских изображений : монография / Р. А. Томакова, С. Г. Емельянов, С. А. Филист ; Юго-Западный государственный университет. - Курск : ЮЗГУ, 2012. - 222 с. - Текст : электронный.

6. Томакова, Римма Александровна . Теоретические основы и методы обработки и анализа микроскопических изображений биоматериалов : монография / Р. А. Томакова, С. А. Филист, С. Г. Емельянов ; МИНОБРНАУКИ РФ, Юго-Западный государственный университет. - Курск : ЮЗГУ, 2011. - 202 с. - Текст : электронный.

7. Методы цифровой обработки изображений : учебное пособие / А. Е. Архипов, С. В. Дегтярев, С. С. Садыков, С. Н. Середа, В. С. Титов. - Курск : КГТУ, 2002 - Ч. 2. - 115 с. - Текст : непосредственный.

9. Технологии сетей связи. Особенности кодирования цифровых факсимильных сообщений : учебное пособие / А. И. Атакищев [и др.] ; Министерство образования Российской Федерации, Курский государственный технический университет. - Курск : КурскГТУ, 2002.

- 159 с. :ил.табл. - Имеется электрон. аналог. - ISBN 5-7681-0111-X : 62.00 р. - Текст : непосредственный.

10. Гонсалес, Р. Цифровая обработка изображений: практические советы : монография / Р. Гонсалес, Р. Вудс ; пер. П. А. Чочиа, Л. И. Рубанова. – 3-е изд., испр. и доп. – Москва : Техносфера, 2012. – 1104 с. :– (Мир цифровой обработки).– URL: <https://biblioclub.ru/index.php?page=book&id=233465> (дата обращения: 23.03.2022).

11. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. - М. : Техносфера, 2006. - 1072 с. - (Мир цифровой обработки). - ISBN 5-94836-028-8 : 394.66 р. - Текст : непосредственный.

12. Фисенко, В.Т. Компьютерная обработка и распознавание изображений / В.Т. Фисенко, Т.Ю. Фисенко. – СПб.: СПбГУ ИТМО, 2008. – 192 с.

13. Гонсалес, Р. Цифровая обработка изображений в среде Matlab / Р. Гонсалес, Р. Вудс, С. Эддинс. – М.: Техносфера, 2006. – 616 с.