Министерство науки и высшего образования Российской Федерации
Министерство науки, высшего образования и инноваций Кыргызской Республики

Межгосударственная образовательная организация высшего образования
Кыргызско-Российский Славянский университет имени первого Президента Российской Федерации Б.Н. Ельцина.
Политехнический институт
Высшая школа цифровых технологий

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

СИСТЕМЫ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

Уровень высшего образования
БАКАЛАВРИАТ

Направление подготовки
01.03.02 - РФ, 510200 - КР, Прикладная математика и информатика
(код и наименование направления подготовки)

Квалификация
Бакалавр

Бишкек 2026

Фонд оценочных средств предназначен для контроля знаний обучающихся по направлению подготовки (специальности) « Прикладная математика и информатика» по дисциплине

«СИСТЕМЫ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ»

наименование

Фонд оценочных средств рассмотрен и утвержден на заседании высшей школы

Цифровые технологии

наименование высшей школы

протокол №1 от «02» 08. 2026г.

Директор высшей школы Цифровых технологий



Хмелева Ирина Владимировна

наименование кафедры *подпись* *расшифровка подписи*

Руководитель образовательной программы

Доулбекова Салтанат Байызбековна Доцент

ФИО *должность* *подпись*

Исполнители:

Доцент Аширбаев Бейшембек Ыбышевич

расшифровка подписи подпись

Профессор Керимбеков Акылбек Керимбекович

расшифровка подписи подпись

Раздел 1. Перечень компетенций, с указанием этапов их формирования в процессе освоения образовательной программы

Формируемые компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций	Виды оценочных средств/ шифр раздела в данном документе
ПК-3. Способность разрабатывать требования и осуществлять проектирования программного обеспечения, баз данных и программных интерфейсов	**Знать:** - типы архитектуры программного обеспечения и её построения; - используемые типовые решения, библиотеки программных модулей, классы объектов при разработке программного обеспечения; - методы и средства проектирования программного обеспечения, баз данных и программных интерфейсов.	Блок А: СРС – лабораторная работа
	Уметь: - использовать современные шаблоны проектирования программного обеспечения и типовые решения; - применять методы и средства проектирования программного обеспечения и программных интерфейсов, осуществлять их тестирования; - применять методы и средства разработки распределенных баз данных и больших структур данных.	Блок Б: Тесты, контрольные вопросы
	Владеть: - навыками разработки, изменения и тестирования программного обеспечения баз данных и программных интерфейсов; - навыками осуществлении коммуникационной деятельности между участниками проектов; - методами проектировании распределенных баз данных и больших структур данных.	

Раздел 2. Технологическая карта дисциплины

Технологическая карта дисциплины

«Системы цифровой обработки сигналов»

Курс/семестр: 4/8

Количество кредитов(ЗЕ): 3

Отчетность: Зачет с оценкой

Название модулей дисциплины согласно РПД	Контроль	Форма контроля	зачетный минимум	зачетный максимум	график контроля
Модуль 1					
Типы изображений и работа с файлами изображений	Текущий контроль	Групповая дискуссия, тестирование	5	10	30
	Рубежный контроль	Практическое занятие, индивидуальные задания	5	10	
Модуль 2					
Амплитудные преобразовани я изображения	Текущий контроль	Групповая дискуссия, тестирование	5	10	35
	Рубежный контроль	Практическое занятие, индивидуальные задания	5	10	
Модуль 3					
Функции спектрального анализа изображения. Преобразование Фурье	Текущий контроль	Групповая дискуссия, тестирование	10	15	40
	Рубежный контроль	Практическое занятие, индивидуальные задания	10	15	
ВСЕГО за семестр			40	70	
Промежуточный контроль (Зачет с оценкой)		Контрольные вопросы.	20	30	
Семестровый рейтинг по дисциплине			60	100	

Модуль	логически завершенная часть дисциплины

Текущий контроль	самостоятельная работа студента, посещаемость и активность на занятиях
Рубежный контроль	проверка полноты знаний и умений по материалу модуля в целом
Промежуточный контроль	завершенная задокументированная часть учебной дисциплины - совокупность тесно связанных между собой модулей дисциплины.

Раздел 3. Типовые контрольные задания и иные материалы, необходимые для оценки планируемых результатов обучения по дисциплине. Описание показателей и критериев оценивания компетенций, описание шкал оценивания.

Блок А:

ЛАБОРАТОРНАЯ РАБОТА. ФУНКЦИИ ФИЛЬТРАЦИИ ИЗОБРАЖЕНИЯ

Цель работы – изучение функций зашумления и фильтрации изображения, приобретение практических навыков их использования.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Функция добавления шума imnoise

Синтаксис

ID = imnoise(Is, type, params)

Функция imnoise создает новое изображение ID путем добавления шума к исходному полутоновому изображению Is. Функция может добавлять шум трех типов, которые задаются строковой константой type. Количество и смысл параметров params определяются выбранным типом шума. Если параметры не указаны, то будут использованы значения по умолчанию:

1) ID = imnoise(Is, „gaussian“, m,v) добавляет к изображению Isгауссовый белый шум с математическим ожиданием m и дисперсией v. По умолчанию m = 0, v = 0.01;

2) ID = imnoise(Is, „salt&pepper“, d) добавляет к изображению Is импульсный (точечный) шум. Параметр d определяет плотность шума и равен доле искаженных пикселов. По умолчанию d = 0.05;

3) ID = imnoise(Is, „spekle“,v) добавляет к изображению Is мультипликативный шум ID = Is+n*Is, где n – равномерно распределенный случайный шум с математическим ожиданием ноль и дисперсией v. По умолчанию v = 0.04.

Задание 1. Зашумить изображение гауссовым, импульсным и мультипликативным шумом.

[S,map] = imread('c:\Image\ris.bmp');

```
I = ind2gray(S,map);
S1 = imnoise(I,'gaussian',0,0.1);
S2 = imnoise(I,'salt& pepper');
S3 = imnoise(I,'speckle');
figure,subplot(1,3,1),subimage(S1);
subplot(1,3,2),subimage(S2);
subplot(1,3,3),subimage(S3)
```

Фильтрация изображения

Для выполнения линейной пространственной фильтрации используют метод двухмерной пространственной свертки локальной окрестности обрабатываемого элемента с линейным оператором, которая называется маской или матрицей коэффициентов фильтра.

Алгоритм свертки заключается в том, что маска сканирует исходное изображение. На каждом шаге находится сумма произведений элементов маски и соответствующих элементов исходного изображения, и найденное значение присваивается одному элементу результирующего изображения. Достигнувтаким образом конца строки, маска смещается на одну строку вниз, в начало строки, и процесс повторяется.

Имеется две тесно связанные концепции, которые необходимо понимать при совершении линейной пространственной фильтрации. Первая – это корреляция, а вторая – свертка. Корреляция состоит в прохождении маски по изображению. С точки зрения механики процесса, свертка делается так же, но маску надо повернуть на 180° перед прохождением по изображению. Если сдвигаемая маска является симметричной, то корреляция и свертка дают одинаковые результаты.
Ниже приводятся функции, используемые для фильтрации изображения.

Функция вычисления двумерной свертки conv2
Синтаксис
D = conv2(S,h, shape)

Функция выполняет двумерную пространственную свертку изображения S с маской h. Параметр shape, определяющий размер результирующего изображения D, может принимать следующие значения:
1) „full“ – полноразмерная свертка (по умолчанию);
2) „same“ – центральная часть размера изображения S;
3) „valid“– центральная часть размера изображения S с вычетом размера маски.

Функция вычисления двумерной линейной фильтрации filter2
Синтаксис
D = filter2(h,S, shape)

Результат вычисляется как корреляция массива S двумерным фильтром, коэффициенты которого сведены в матрицу h. Как правило, S и D являются полутоновыми изображениями. Значения параметра shape такие же, как у функции conv2. По умолчанию shape = „same“. С точки зрения выполнения процесса, свертка делается так же, но маску h надо повернуть на 180° перед прохождением по изображению S. Для этого можно использовать функцию rot90.

Задание 2. Выполнить фильтрацию палитрового изображения, используя маску

$$h = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{vmatrix}$$

```
 [x,map] = imread('c:\image\Ris.bmp');
I = ind2gray(x,map);
I = im2double(I);
h = [1 1 1;1 -2 1;-1 -1 -1];
h1 = rot90(h,2);
I1 = filter2(h1,I);
figure,imshow(I)
figure,imshow(I1)
```

Функция вычисления медианной фильтрации medfilt2

Синтаксис

ID = medfilt2(Is,[m,n], padopt)

Функция выполняет нелинейную фильтрацию, механизм выполнения которой аналогичен линейной фильтрации, но на каждом шаге сканирования маски размера mxn (по умолчанию 3 3) выполняется следующая нелинейная операция:

пикселы изображения, находящиеся под маской, сортируются и составляют упорядоченную последовательность A. Пикселу результирующего изображения ID (r,c), где r и c – координаты текущего положения центрального элемента маски, присваивается значение медианы последовательности A.

Параметр ***padopt*** определяет три возможные опции расширения границ изображения:

'zeros'опция по умолчанию с нулевым расширением,

'symmetric', при которой изображение Is расширяется путем его зеркального отражения через границы, и

'indexed', при которой Is расширяется значением 1, если Is имеет класс double, и значением 0 в противном случае.

Задание 3. Отфильтровать изображение файла Ris.bmp, зашумленного импульсным шумом.

```
[S,map] = imread('c:\Image\Ris.bmp');
I = ind2gray(S,map);
S = imnoise(I,'salt& pepper');
figure,imshow(S,[]);
D = medfilt2(S);
figure,imshow(D,[]);
```

Функция вычисления двумерной АЧХ freqz2

Синтаксис

freqz2(h,n1,n2) – выводит на экран двумерную АЧХ.

[H,f1,f2] = freqz2(h,n1,n2) – формирует матрицу H размера n1 n2, которая является АЧХ на частотах, содержащихся в векторах f1 и f2, по двумерному фильтру, коэффициенты которого сведены в матрицу h.

Задание 4. Вывод АЧХ высокочастотного фильтра двумя способами.

```
h = [–1,–1,–1;–1,9,–1;–1,–1,–1];
figure,freqz2(h);
title(„1 variant)
```

```
[H,f1,f2] = freqz2(h);
figure,mesh(f1,f2,abs(H))
title(„2 variant)
```

Функция задания маски предопределенного фильтра fspecial

Синтаксис

h = fspecial (type, P1, P2)

Функция возвращает маску h предопределенного двумерного линейного фильтра, задаваемого строкой type. В зависимости от типа фильтра, для него могут быть определены один или два дополнительных параметра P1, P2.

Может быть задан размер маски n:

если n – вектор, то размер маски n(1) × n(2),

если n – скаляр, то размер маски – n × n),

sigma – среднеквадратичное отклонение распределения Гаусса, которое используется при формировании маски h;

параметр a, управляющий соотношением между центральным и граничными элементами маски, устанавливается в диапазоне [0, 1], по умолчанию равен 0.2.

Возможные варианты функции ***fspecial:***

1) h = fspecial ('gaussian', n,sigma) возвращает маску h фильтра нижних частот Гаусса. По умолчанию n равно 3 × 3, a sigma равно 0.5;

2) h = fspecial ('sobel') возвращает маску фильтра Собела для выделения горизонтальных границ, для выделения вертикальных границ достаточно транспонировать данную маску h;

3) h = fspecial ('prewitt') возвращает маску фильтра Превитта для выделения горизонтальных границ, для выделения вертикальных границ достаточно транспонировать данную маску h;

4) h = fspecial ('1ap1acian', a) возвращает маску h ВЧ фильтр Лапласа. Размер маски 3×3. С помощью фильтра Лапласа можно выполнять улучшение изображения, используя формулу

$$g(x,y) = f(x,y) + c\Delta^2 f(x.y)$$

где f(x, y) – это исходное изображение;

g(x, y) – улучшенное изображение;

параметр c = 1, если центральный коэффициент маски положителен, и

c = –1 в противном случае;

отфильтрованное изображение фильтром Лапласа:

$$\Delta^2 f(x.y) = \frac{d^2 f(x,y)}{dx^2} + \frac{d^2 f(x,y)}{dy^2}$$

5) h = fspecial ('log',n,sigma) возвращает маску h фильтра, аналогичного последовательному применению фильтров Гаусса и Лапласа, так называемого лапласиана – гауссиана, n и sigma по умолчанию устанавливается равным 5 × 5 и 0.5 соответственно;

6) h = fspecial ('average', n) возвращает маску h усредняющего НЧ фильтра. По умолчанию размер маски n устанавливается равным 3 × 3;

7) h = fspecial ('unsharp', a) возвращает маску h фильтра, повышающего резкость изображения. Размер маски: 3 × 3.

Задание 5. Улучшить изображение, используя фильтр высоких частот Лапласа.

1. Создать маску h фильтра высоких частот Лапласа:
```
h = fspecial('laplacian',0);
```

2. Выполнить высокочастотную фильтрацию изображения:
```
[S,map] = imread('c:\Image\Ris.bmp');
I = im2double(ind2gray(S,map));
figure,imshow(I)
I1 = filter2(h,I);
figure,imshow(I1)
I2 = I – I1;
figure,imshow(I2)
```

Задание 6. Отфильтровать изображение фильтром Собела.

1. Создать маску фильтра Собеля для выделения горизонтальных границ
```
h = fspecial('sobel');
```

2. Создать маску фильтра Собеля для выделения вертикальных границ
```
h1 = h';
```

3. Отфильтровать изображениеRis.bmp фильтрами Собеля.
```
[X,map] = imread('c:\Image\earth.bmp');
I = ind2gray(X,map);
figure,imshow(I)
F1 = filter2(h,I);
figure,imshow(F1)
F2 = filter2(h1,I);
figure,imshow(F2)
F3=F1+F2;
figure,imshow(F3)
```

Функция формирования маски линейного фильтра по желаемой АЧХ fsamp2
Синтаксис
```
h = fsamp2(f1,f2,H)
```

Функция формирует маску h линейного двумерного фильтра, основываясь на желаемой АЧХ двумерного фильтра H для частот, передаваемых в векторах f1 и f2.
Задание 7. Повысить резкость изображения с помощью линейного фильтра, сформированного по желаемой АЧХ.

1. Получить нормализованные значения частот:
```
[f1,f2] = freqspace(15,'meshgrid');
```

2. Вычислить желаемую АЧХ - H, используя так называемую метрику городских кварталов:
```
dist = abs(f1)+abs(f2);
H = dist/max(dist(:));
```

3. Вывести на экран желаемую АЧХ:
```
figure,mesh(f1,f2,H),colormap(cool(32));
```

4. Сформировать маску фильтра по желаемой АЧХ:
h = fsamp2(f1,f2,H,[5 5]);

5. Вывести на экран получившуюся АЧХ:
figure,colormap(cool(32)),freqz2(h);

6. Прочитать изображение и вывести его на экран:
[i,map] = imread('c:\Image\Ris.bmp');
figure,imshow(i,map);

7. Отфильтроватьизображение:
i1 = mat2gray(filter2(h,im2double(i)));
figure,imshow(i1);

8. Контрастирование результата фильтрации:
i2 = imadjust(i1,[0 0.5],[]);
figure,imshow(i2);

Функция формирования маски линейного фильтра методом преобразования частот ftrans2

Синтаксис
h = ftrans2(b)

Функция формирует маску h линейного двумерного фильтра, используя метод преобразования частот для трансформации одномерного фильтра с коэффициентами b. Для преобразования частот используется специальная матрица трансформации t. По умолчанию применяется матрица Мак–Клеллана:

$$t = \frac{1}{8} \cdot \begin{vmatrix} 1 & 2 & 1 \\ 2 & -4 & 2 \\ 1 & 2 & 1 \end{vmatrix}$$

В функции h = ftrans2(b,t) можно указать иную матрицу трансформации.

Задание 8. Выполнить размытие изображения с целью подавления муара с помощью линейного двумерного фильтра низкой частоты (ФНЧ), сформированного из одномерного ФНЧ методом преобразования частот:

1. Создать одномерный ФНЧ 14-го порядка с частотой среза 0.2:
b = fir1(14,0.2); % проектирование фильтров методом взвешивания
freqz(b,1,256)

2. Сформировать двумерный фильтр из одномерного:
h = ftrans2(b);
figure,freqz2(h)

3. Прочитать исходное изображение и отфильтровать полученным фильтром:
[S,map] = imread('c:\Image\Ris.bmp');
I = ind2gray(S,map);
I = im2double(I);
figure,imshow(I)
I1 = filter2(h,I);
figure,imshow(I1)
I2 = mat2gray(I1);

```
figure,imshow(I2)
```

Функция вычисления общей нелинейной фильтрации nlfilter
Синтаксис
D = nlfilter(S,[m n],fun,P1,P2, …)

Функция D = nlfilter(S,[m n],fun,P1,P2, …) используется для обработки бинарных и полутоновых изображений. Формирует новое изображение, используя для фильтрации маску размером [m n] и функцию, осуществляющую фильтрацию, fun. P1,P2 – возможные параметры, передаваемые в fun.

Задание 9. Создать функцию по фильтрации импульсного шума для использования в качестве аргумента fun в функции обобщенного нелинейного фильтра, который в этом случае будет использоваться для удаления импульсного шума в изображении. В функции фильтрация импульсного шума осуществляется операцией усреднения с порогом, которая состоит в том, что центральному пикселу в пределах маски присваивается среднее значение яркости всех пикселов в пределах маски в том случае, если разница между исходным значением центрального пиксела и средним больше заданного порогового значения. Для создания этой функции с именем 'AverageWithTh' в редакторе m-файлов необходимо выполнить следующие действия.

1. Создать новый m-файл
2. Ввести команды функции:

```
functionR = AverageWithTh(x,Th); % заголовочнаястрока
[r c] = size(x);
n = r*c;
r = floor((r+1)/2); c = floor((c+1)/2);
s = sum(x(:))/n;
if (abs(x(r,c)–s))>Th
R = s;
else
R = x(r,c);
end;
end;
```

3. Прочитать изображение, зашумить импульсным шумом и отфильтровать его, используя созданную функцию:

```
[X,map] = imread('c:\Image\Ris.bmp');
I = im2double(ind2gray(X,map));
figure,imshow(I)
I1 = imnoise(I,'salt& pepper');
figure,imshow(I1)
I2 = nlfilter(I1,[3 3],'AverageWithTh',0.2);
figure,imshow(I2)
```

Блок Б:

Приложение Контрольные вопросы.

Знать:

1. Какие классы данных (форматы) представления пикселов изображения существуют?
2. Какие типы растровых изображений используются в пакете IPT?

3. С помощью какой функции можно получить информацию о размере, типе изображения?
4. С какими форматами графических файлов можно работать в системе MatLab?
5. С помощью каких функций можно прочитать изображение из файла на диске и записать изображение на диск?
6. Какие аргументы функции imshow изменяют контраст полутонового изображения при его выводе на экран?
7. Какие вы знаете функции преобразования типов изображений?
8. Каким образом осуществляется дискретизация сигнала?
9. Как выбирается величина шага дискретизации?
10. Каким образом осуществляется квантование сигнала?
11. Для чего используется функция а) quantiz; б) blkproc?
12. Какими способами можно задать вырезание фрагмента для функции imcrop?
13. Что является входными аргументами для функции изменения размеров?
14. Что является входными аргументами для функции поворота?
15. Как задать аффинную матрицу:
16. а) для растяжения; б) для сдвига; в) для поворота; г) для переноса?
17. Как можно выполнить зеркальное отображение изображения?
18. Как можно выполнить «прореживание» изображения?
19. В чем особенность амплитудных преобразований?
20. Что такое контраст изображения?
21. Какие входные аргументы передаются в функции по изменению контраста?
22. Какие вы знаете способы вызова функции brighten?
23. С помощью, какой функции можно уменьшить количество цветов палитрового изображения? Какие входные аргументы используются для этого?
24. Что такое гистограмма?
25. Какая функция используется для получения гистограммы?
26. В чем отличие гистограммы полутонового изображения от гистограммы палитрового изображения?
27. Что такое эквализация изображения? Какая функция выполняет эквализацию? Ее способы вызова.
28. В чем смысл корреляционного анализа сигналов?
29. Какие функции корреляционного анализа вы знаете?
30. Какие типы фильтров создает функция по формированию масок фильтров fspecial?
31. В чем заключается алгоритм двумерной свертки?
32. В каких функциях присутствует алгоритм двумерной свертки?
33. В чем отличие алгоритма медианной фильтрации от алгоритма фильтрации с помощью операции усреднения с порогом?
34. Какие типы шумов формирует функция по зашумлению изображений imnoise?
35. Для каких целей можно использовать функцию freqz2?
36. Каким образом можно сформировать маску линейного фильтра по желаемой АЧХ?
37. Какая функция позволяет сформировать двумерный фильтр из одномерного?
38. Что обуславливает искажения изображения при его формировании?
39. В чем заключается задача построения восстанавливающей системы?
40. Какие функции использовались для моделирования «размытия» изображения и его восстановления в задании лабораторной работы?
41. Какие принципы лежат в основе построения фильтров Винера, гомоморфного фильтра?
42. Какие логические операции над бинарными изображениями вы знаете?
43. В чем назначение структурообразующего элемента в морфологических операциях?
44. Для чего используются морфологические операции?
45. Какие морфологические операции обработки изображения относятся к базовым?

46. Какие операции являются комбинированием эрозии и дилатации?
47. Какие функции пакета IPT выполняют операции эрозии и дилатации, замыкания, размыкания?
48. В чем заключается сегментация изображения?
49. Какие признаки используются для сегментации?
50. В чем заключается метод выращивания областей, использующийся для сегментации изображения?
51. В чем заключается метод разделения, использующийся для сегментации изображения?
52. Что является входными параметрами функции сегментации методом разделения?
53. В чем заключается преобразование яркостного среза?
54. Какие параметры возвращает функция impixel?
55. Как выделить нужные объекты изображения с помощью функции bwselect?
56. Что из себя представляет матрица номеров объектов?
57. Какие признаки вычисляет функция imfeature?
58. Какие виды изображений объекта можно получить с помощъю функции imfeature?
59. Какие типы габаритов объекта можно определить с помощью функции imfeature?
60. Какие типы площадей объекта можно определить с помощью функции imfeature?
61. Какие коэффициенты формы объекта можно определить с помощью функции imfeature?
62. Какие вычисляемые функцией imfeature признаки относятся к оставшимся?
63. Какие функции используются для выполнения двумерного прямого и обратного преобразования Фурье в системе MatLab?
64. Зачем используется двумерная дискретизация? Приведите примеры функции дискретизации.

Уметь и владеть:

1. Получить информацию об изображении с расширение.bmp.
2. Прочитать изображение, преобразовать его в изображение:
   2.1. полутоновое
   2.2. палитровое.
3. Полутоновое изображение вывести на экран, дополнительно отконтрастировав таким образом, чтобы пикселы, яркость которых меньше или равна 0.2, отображалась черным цветом, а пикселы, яркость которых больше или равна 0.8 – белым.
4. Преобразовать пикселы полноцветного и полутонового изображений в формат double.
5. Вывести полноцветное и полутоновое изображения на экран в одно окно, разместив их по вертикали.
6. Выполнить преобразования изображения:

№ задания	Преобразования изображений
1.	Получить информацию из файла об изображении
2.	Прочитать изображение из файла
3.	Вырезать квадратный фрагмент D с помощью мыши
4.	Вырезать фрагмент, задавая в команде положение координаты его верхнего левого угла как целую часть от 1/3 ширины и высоты изображения соответственно, значения ширины и высоты фрагмента определяются также
5.	Вырезать фрагмент с помощью мыши из текущего окна без задания для него переменной и из полученного окна вырезать квадратный фрагмент с помощью мыши в переменную S
6.	Увеличить фрагмент в (N) раз, использовав метод: a – 'nearest'; b – „bilinear“; d – 'bicubic'

7.	Повернуть изображение на заданный угол по часовой стрелке (угол), использовав метод: a – 'nearest'; b – „bilinear“; d –'bicubic'
8.	Повернуть изображение на заданный угол против часовой стрелки (угол), использовав метод: a – 'nearest'; b – „bilinear“; d – 'bicubic'
9.	Уменьшить изображение в (N) раз, использовав метод: a – 'nearest', b – „bilinear“, d – 'bicubic'
10.	Растянуть изображение по горизонтали и по вертикали в (Sx, Sy) раз
11.	Скомбинировать сдвиг изображения по горизонтали и по вертикали на (a, b)
12.	Перенести изображение по горизонтали и по вертикали на (Sx, Sy)
13.	Вывести на экран исходное изображение
14.	Вывести на экран результаты преобразования в одно окно

Выполнить следующие действия.

7. Прочитать палитровое изображение из файла Ris1.bmp, вывести на экран, преобразовать в полутоновое, добавить импульсный шум, отфильтровать зашумленное изображение медианной фильтрацией и обобщенным нелинейным фильтром. Зашумленное и отфильтрованные изображения вывести в одном окне для сравнения.
8. Выполнить фильтрацию изображения из файла Ris2.bmp:
   a) Отфильтровать изображение с помощью масок кругового градиента

север			северо-восток			восток			юго-восток		
1	1	1	1	1	1	−1	1	1	−1	−1	1
1	−2	1	−1	−2	1	−1	−2	1	−1	−2	1
−1	−1	−1	−1	−1	1	−1	1	1	1	1	1

юг			юго-запад			запад			северо-запад		
−1	−1	−1	1	−1	−1	1	1	−1	1	1	1
1	−2	1	1	−2	−1	1	−2	−1	1	−2	−1
1	1	1	1	1	1	1	1	−1	1	−1	−1

Ввод масок двумерных линейных фильтров, построение их АЧХ, фильтрацию изображения с их помощью и вывод результата выполнить в цикле;

   b) Улучшить изображение с помощью масок лапласиановских фильтров:

0	1	0	−1	−1	−1	1	−2	1
1	−4	1	−1	8	−1	−2	4	−2
0	1	0	−1	−1	−1	1	−2	1

Ввод масок и вывод результатов выполнить в цикле.

9. Прочитать изображение из файла Ris.bmp, вывести на экран, преобразовать в полутоновое. Получить маску фильтра Превитт. Выполнить фильтрацию исходного полутонового фрагмента маской фильтра Превитт отдельно по горизонтали и по вертикали и вместе на одном изображении.
10. Прочитать цветное изображение из файла 'Ris3.bmp', вывести на экран, преобразовать в полутоновое. Взять в качестве АЧХ фильтра функцию расстояния от начала координат, сформировать по АЧХ маску фильтра и отфильтровать полутоновое

изображение. Исходное полутоновое изображение и результаты его обработки вывести на экран.

11. Вывести АЧХ всех фильтров, создаваемых функцией по заданию масок предопределенного фильтра, в одном окне с заголовками для каждого фильтра.
12. Зашумить в цикле полутоновое изображение гауссовым шумом с разными дисперсиями: а) по умолчанию (0.01); б) v = 0.5 для математического ожидания, равного m = 0.5.

13. Реализовать алгоритм, выполняющий операцию свертки исходного изображения с матрицей-маской. Используйте следующие матрицы маски для реализации различных алгоритмов обработки изображений:

- для сглаживания и подавления шумов в изображении используют матрицу-маску размером 3x3 следующего вида:

$$1/9*\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- для подчеркивания контуров используются матрицы-маски следующего вида:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$1/9*\begin{bmatrix} -1 & -1 & -1 \\ -1 & 17 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

- для выделения контуров используются маска следующего вида:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

14. Реализовать алгоритм тиснения. Тиснение делается аналогично алгоритмам усреднения или подчеркивания контуров. Каждый пиксель в изображении обрабатывается ядром (матрицей-маской) тиснения размером 3x3. Например, в качестве ядра тиснения можно взять следующую матрицу-маску:

$$\begin{matrix} 0 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 0 \end{matrix}$$

После того, как значение пиксела обработано ядром тиснения, к нему прибавляется 128. Таким образом значением фоновых пикселов станет средний серый цвет (красный = 128, зеленый = 128, синий = 128). Суммы, превышающие 255, можно округлить до 255.

В тисненом варианте изображения контуры кажутся выдавленными над поверхностью. Направление подсветки изображения можно изменять, меняя позиции 1 и -1 в ядре. Если, например, поменять местами значения 1 и -1, то реверсируется направление подсветки.

15. Акварелизация изображения. Акварельный фильтр преобразует изображение, и после обработки оно выглядит так, как будто написано акварелью:

- первый шаг в применении акварельного фильтра - сглаживание цветов в изображении. Одним из способов сглаживания является применение медианного усреднения цвета в каждой точке. Значение цвета каждого пикселя и его 24 соседей (размер матрицы-маски равен 5x5) выстраиваются в вариационный ряд по убыванию или возрастанию. Медианное (тринадцатое) значение цвета в вариационном ряде присваивается центральному пикселю.
- после сглаживания цветов необходимо применить фильтр подчеркивания контуров, чтобы выделить границы переходов цветов.
- 1. Выполнить бинаризацию изображения файла Construc.bmp с использованием автоматического выбора порога.
- 2. Выполнить бинаризацию изображения файла Construc.bmp.
- 3. Получить негатив изображения файла Athena.bmp

Приложение 2

ШКАЛА ОЦЕНИВАНИЯ ПИСЬМЕННОГО ОПРОСА
(промежуточный контроль – «ЗНАТЬ»)

- Отметкой **(9-10 баллов)** оценивается ответ, который показывает прочные знания основных приемов программирования в пакета Матлаб, математических основ применяемых алгоритмов обработки изображения.
- Отметкой **(7-8 баллов)** оценивается ответ, обнаруживающий прочные знания основных приемов программирования в пакета Матлаб, математических основ применяемых алгоритмов. Однако допускается одна - две неточности в ответе.
- Отметкой **(4-6 баллов)** оценивается ответ, свидетельствующий в основном о знании основных понятий основных приемов программирования в пакета Матлаб, математических основ применяемых алгоритмов. Допускается несколько ошибок в содержании ответа.
- Отметкой **(2-3 балла)** оценивается ответ, обнаруживающий незнание основных приемов программирования в пакета Матлаб, математических основ применяемых алгоритмов. Отмечается отсутствие логичности и последовательности в ответе. Допускаются серьезные ошибки в содержании ответа.
- Отметкой **(0 -1 балл)** оценивается ответ, при котором студент демонстрирует непонимание проблемы или нет ответа.

ШКАЛА ОЦЕНИВАНИЯ ПРАКТИЧЕСКИХ ЗАДАНИЙ
(промежуточный контроль – «УМЕТЬ и ВЛАДЕТЬ»)

- Отметкой **(17-20 баллов)** оценивается ответ, при котором студент правильно решает индивидуальные задачи по предложенному алгоритму. Демонстрирует способность грамотно формализовать задачу из индивидуального задания.
- Демонстрирует полное понимание проблемы. Все требования, предъявляемые к заданию, выполнены.
- Отметкой **(11-16 баллов)** оценивается ответ, при котором студент в основном правильно решает индивидуальные задачи по предложенному алгоритму, демонстрирует способность грамотно формализовать задачу из индивидуального задания. Демонстрирует значительное понимание проблемы. Большинство требований, предъявляемых к заданию, выполнены.
- Отметкой **(4-10 баллов)** оценивается ответ, при котором студент в основном не правильно решает индивидуальные задачи по предложенному алгоритму, демонстрирует способность грамотно формализовать задачу из индивидуального задания. Демонстрирует частичное или небольшое понимание проблемы. Многие требования, предъявляемые к заданию, не выполнены.
- Отметкой **(0 -3 балла)** оценивается ответ, при котором студент демонстрирует непонимание проблемы или нет ответа и даже не было попытки решить задачу

Раздел 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Изучение дисциплины студентами осуществляется в форме лекций, практических занятий в аудиторных условиях (компьютерные классы) и в процессе самостоятельной работы, контроля знаний.

Теоретическая информация, по мере возможности, представляется в виде компьютерных презентаций с использованием мультимедийных средств.

Практические занятия проводятся в компьютерных классах, оснащенных персональными компьютерами необходимыми параметрами и с установленным необходимым программным обеспечением. Используется Интернет для получения дополнительной информации. Используется дискуссионный метод проведения занятий, где студенты могут высказать свое мнение по обсуждаемой проблеме.

Защита практических работ проводится в виде собеседования с преподавателем по теории и программной реализации работы.

Перечень информационно справочных систем и программного обеспечения

операционная система Microsoft Windows, пакет программ Microsoft Office 2010, языки программирования высокого уровня, установленные на компьютерах, учебно-методические комплексы по разделам дисциплины, размещенные на серверах компьютерных классов.

Раздел 5. Методические указания для обучающегося по освоению дисциплины. Математические методы решения прикладных задач.

Текущий контроль осуществляется в течение семестра и включает в себя проверку подготовки студента к выполнению
практической работы (т.е. знание теории по теме практической работы), проверку завершения практической работы и
оформления отчёта.
Рекомендации по организации самостоятельной работы студента
1. Советы по планированию и организации времени, необходимого для изучения дисциплины. Рекомендуется следующим
образом организовать время, необходимое для изучения дисциплины:
Изучение конспекта лекции в тот же день, после лекции – 10-15 минут.
Изучение конспекта лекции за день перед следующей лекцией – 10-15 минут.
Изучение теоретического материала по учебнику и конспекту – 1 час в неделю.
Подготовка к практическому занятию – 2 час.
Всего в неделю – 3 часа 30 минут.
2. Описание последовательности действий студента
Для понимания материала и качественного его усвоения рекомендуется такая последовательность действий:
2.1. После прослушивания лекции и окончания учебных занятий, при подготовке к занятиям следующего дня, нужно
сначала просмотреть и обдумать текст лекции, прослушанной сегодня (10-15 минут).
2.2. При подготовке к лекции следующего дня, нужно просмотреть текст предыдущей лекции, подумать о том, какая может
быть тема следующей лекции (10-15 минут).
2.3. В течение недели выбрать время (1-час) для работы с рекомендуемой литературой в библиотеке.
2.4. При подготовке к практическим занятиям следующего дня, необходимо сначала прочитать основные понятия и подходы
по теме домашнего задания. При выполнении упражнения или задачи нужно сначала понять, что требуется в задаче, какой
теоретический материал нужно использовать, наметить план решения задачи.
3. Рекомендации по использованию материалов учебно-методического комплекса. Рекомендуется использовать
методические указания по курсу, текст лекций преподавателя.

4. Рекомендации по работе с литературой. Теоретический материал курса становится более понятным, когда дополнительно
к прослушиванию лекции и изучению конспекта, изучаются и книги. Легче освоить курс, придерживаясь одного учебника и
конспекта. Рекомендуется, кроме «заучивания» материала, добиться состояния понимания изучаемой темы дисциплины. С
этой целью рекомендуется после изучения очередного параграфа выполнить несколько простых упражнений на данную
тему. Кроме того, очень полезно мысленно задать себе следующие вопросы (и попробовать ответить на них): о чем этот
параграф?, какие новые понятия введены, каков их смысл?, что даст это на практике?.

5. Советы по подготовке к рубежному и промежуточному контролям. Дополнительно к изучению конспектов лекции
необходимо пользоваться учебником. Кроме «заучивания» материала, очень важно добиться состояния понимания
изучаемых тем дисциплины. С этой целью рекомендуется после изучения очередного параграфа выполнить несколько
упражнений на данную тему. Кроме того, очень полезно мысленно задать себе

следующие вопросы (и попробовать ответить
на них): о чем этот параграф?, какие новые понятия введены, каков их смысл?, что даст это на практике?.
При подготовке к промежуточному контролю нужно изучить теорию: определения всех понятий и подходы к оцениванию
до состояния понимания материала и самостоятельно решить несколько типовых задач из каждой темы. При решении задач
всегда необходимо уметь качественно интерпретировать итог решения.

6. Указания по организации работы с контрольно-измерительными материалами, по выполнению домашних заданий. При
выполнении домашних заданий необходимо сначала прочитать основные понятия и подходы по теме задания. При
выполнении упражнения или задачи нужно сначала понять, что требуется в задаче, какой теоретический материал нужно
использовать, наметить план решения задачи, а затем приступить к расчетам и сделать качественный вывод.
Рубежный контроль осуществляется в течение семестра в виде защиты практических работ, выполнения контрольных
работ, тестирования и контрольных вопросов.
Методические указания по выполнению практических работ представлены в электронной папке преподавателя (локальная
сеть кафедры Информационных и вычислительных технологий КРСУ). Выполнение практических работ завершается
оформлением отчета, в котором приводится теория по теме работы, результаты работы программы и графики.
Рекомендации по работе с тестами по дисциплине.
Совокупность тестов, представленная в разделе 5.3 может использоваться по частям для текущего контроля освоения
предмета по разделам и темам и для итогового контроля знаний по дисциплине в целом.
Предполагается «загрузить» тесты в специализированную ПС или использовать их в форме бланочного контроля. Они
могут использоваться студентами для самоконтроля при подготовке к итоговому зачёту по дисциплине.
Вопросы по контрольным работам представлены в разделе РПД Фонд оценочных средств.
Ответы на контрольные работы оформляются в письменном виде.
Список литературы должен быть составлен в полном соответствие с действующим стандартом (правилами).
Для допуска к зачету студент в течении семестра должен сдать все практические работы.