

Лабораторная работа №5: Синтезирование изображений

Цель работы – приобретение навыков синтеза цифровых изображений, генерации шума различной природы возникновения. Использование базовых арифметических и геометрических операций над цифровым изображением.

Краткие теоретические сведения

Под синтезированным цифровым изображением понимается любое визуальное представление информации, получаемое в результате вычислений в некоторой вычислительной системе (компьютере), выполняющей некоторые директивы пользователя (получателя изображения), вне всякой физической связи с реальным изображением. Последнее замечание исключает из предмета цифровой обработки изображений такие задачи, как обработка изображений, получаемых при фотографии или видеосъемке в реальном времени, поскольку в данном случае имеют дело с физическими объектами, а не с построением образов графических объектов по тем или иным их описаниям. Например, описанием окружности, изображение которой надо получить на экране монитора, может служить ее уравнение, или координаты центра и радиус, или двоичное представление ее контура.

Множество видимых на экране элементов, упорядоченных согласно определенным пространственным отношениям (в качестве таких отношений можно, например, назвать отношения «дальше-ближе», «больше-меньше» и т.д.), с заданными атрибутами окружения (например, уровень освещенности, признак наличия дымки и её плотности), с атрибутами поверхностей элементов (цвет, текстура, оптические свойства материалов моделируемых объектов) образуют сцену. Видимые глазом графические объекты, из которых состоит сцена, будем называть элементами сцены.

Одной из главных составляющих на изображении является шум. Шумом называется дефект изображения, вносимый влиянием окружающей среды, фотосенсорами и электроникой устройств, которые их используют, вследствие несовершенства технологий. Шум проявляется на изображении в виде наложенной маски из пикселей случайного цвета и яркости. Синтезирование шума на изображении принято реализовывать на основе заданной функции плотности вероятности или функции распределения. Большинство генераторов случайных чисел в языках программирования (включая MATLAB) основаны на переформулировании задачи в терминах случайных чисел с равномерной функцией распределения в интервале $[0, 1]$ – с помощью функции `rand`. На основе равномерного закона распределения могут быть получены любые виды распределения шумов. В

основе этого подхода лежит одно из утверждений теории вероятности о том, что если имеется случайная величина с равномерным распределением на отрезке $[0, 1]$, то случайную величину z с заданной функцией распределения F_z можно построить по формуле:

$$z = F^{-1}(w),$$

т.е. в эквивалентной форме необходимо решить уравнение вида $F(z) = w$ относительно z .

Например, для генерации случайной величины z с функцией распределения вероятностей Релея, следует воспользоваться следующей формулой:

$$F_z(z) = \begin{cases} 1 - e^{-(z-a)^2/b} & \text{при } z \geq a \\ 0 & \text{при } z < a \end{cases}$$

При использовании случайных чисел с равномерным распределением в интервале $[0, 1]$, чтобы получить z , достаточно решить уравнение:

$$1 - e^{-(z-a)^2/b} = w \quad \text{или} \quad z = a + \sqrt{b \ln(1-w)}$$

Поскольку квадратный корень – положительная функция, то генерируемые случайные величины будут всегда больше a , что требуется в определении функции Релея. Значит, равномерно распределенные случайные числа смогут служить основой для генератора релеевских случайных чисел с заданными параметрами a и b .

Выражение вида $z = a + \sqrt{b \ln(1-w)}$ иногда называют уравнением

генератора случайных чисел, поскольку в нем определяется, как вычислять требуемые случайные величины. В этом случае имеется простая формула для решения уравнения. Однако это не всегда возможно, и проблему можно сформулировать следующим образом: как получить уравнение генератора случайных чисел, выход которого хорошо приближает случайную величину с заданной функцией плотности вероятностей.

В таблице приведены формы представления случайной величины. Для некоторых из них удастся выписать формулу обратной функции распределения, например, для экспоненциального распределения или для функции распределения Релея. В этих случаях имеется простая формула для выражения требуемых случайных чисел в терминах равномерных случайных чисел. В других случаях, как в случае гауссовой или логарифмически нормальной плотности, такой простой формулы не существует. Тогда необходимо искать альтернативный способ – например, для случайной величины z с логарифмически нормальной плотностью

можно воспользоваться тем фактом, что величина $\ln(z)$ имеет гауссово распределение, и выписать выражение в терминах гауссовой случайной величины с нулевым средним

| Имя | Плотность | Среднее и дисперсия | Распределение | Генератор |
|---------------------------------------|---|--|---|---|
| Равно- мерная | $p_z(z) = \begin{cases} \frac{1}{b-a}, & a \leq z \leq b \\ 0, & \text{иначе} \end{cases}$ | $m = \frac{a+b}{2},$ $\sigma^2 = \frac{(b-a)^2}{12}$ | $F_z(z) = \begin{cases} 0, & z < a \\ \frac{z-a}{b-a}, & a \leq z \leq b \\ 1, & z > b \end{cases}$ | rand(MATLAB) |
| Гауссова | $p_z(z) = \frac{1}{\sqrt{2\pi}b} e^{-(z-a)^2/2b^2},$ $-\infty < z < +\infty$ | $m = a,$ $\sigma^2 = b^2$ | $F_z(z) = \int_{-\infty}^z p_z(v) dv$ | randn(MATLAB) |
| «Соль и перец» | $p_z(z) = \begin{cases} P_a, & z = a \\ P_b, & z = a, (b > a) \\ 0, & \text{иначе} \end{cases}$ | $m = aP_a + bP_b,$ $\sigma^2 = (a-m)^2 P_a + (b-m)^2 P_b$ | $F_z(z) = \begin{cases} 0, & z < a \\ P_a, & a \leq z < b \\ P_a + P_b, & z \geq b \end{cases}$ | rand(MATLAB) |
| Логарифм ически норма- льная | $p_z(z) = \frac{1}{\sqrt{2\pi}bz} e^{-(z-a)^2/2b^2}, \quad z > 0$ | $m = e^{a+(b^2/2)},$ $\sigma^2 = \left[e^{b^2} - 1 \right] e^{2a+b^2}$ | $F_z(z) = \int_{-\infty}^z p_z(v) dv$ | $z = ae^{bN(0,1)}$ |
| Релея | $p_z(z) = \begin{cases} \frac{2}{b} (z-a) e^{-(z-a)^2/b}, & z \geq a \\ 0, & z < a \end{cases}$ | $m = a + \frac{2\pi b}{4},$ $\sigma^2 = \frac{b(4-\pi)}{4}$ | $F_z(z) = \begin{cases} 1 - e^{-(z-a)^2/b}, & z \geq a \\ 0, & z < a \end{cases}$ | $z = a + \sqrt{b \ln[1 - U(0,1)]}$ |
| Экспонен- циальная | $p_z(z) = \begin{cases} ae^{-az}, & z \geq 0 \\ 0, & z < 0 \end{cases}$ | $m = \frac{1}{a},$ $\sigma^2 = \frac{1}{a^2}$ | $F_z(z) = \begin{cases} 1 - e^{-az}, & z \geq 0 \\ 0, & z < 0 \end{cases}$ | $z = -\frac{1}{a} \ln[1 - U(0,1)]$ |
| Эрл анга | $p_z(z) = \frac{a^b z^{b-1}}{(b-1)!} e^{-az}, \quad z \geq 0$ | $m = \frac{a}{b},$ $\sigma^2 = \frac{a}{b^2}$ | $F_z(z) = \left 1 - e^{-az} \sum_{n=0}^{b-1} \frac{(az)^n}{n!} \right ,$ | $z = E_1 + E_2 + \dots + E_b,$ $E_n - \text{эксп. величины с}$ $\text{параметром } a$ |

В таблице $N(0,1)$ – нормальная (гауссова) случайная величина со средним ноль и дисперсией 1. $U(0,1)$ – равномерная случайная величина из интервала (0, 1).

Представленные в таблице генераторы случайной величины играют важную роль при моделировании поведения случайного шума в приложениях обработки изображений. Так гауссов шум используется в качестве естественного приближения в тех случаях, когда детекторы изображения работают на пороге чувствительности. Шум типа «соль и перец» возникает в устройствах с ошибочной коммутацией. Размеры зерен на фотоэмульсии являются случайными величинами с логарифмически нормальным законом распределения. Шум Релея образуется при фиксации удаленных изображений, а экспоненциальный шум и шум Эрланга используются при описании искажений на изображениях, полученных с помощью лазерного излучения.

Пример кода функции для генерации различных шумов на изображениях

```
function R = imnoise2(type, M, N, a, b)
%IMNOISE2 Generates an array of random numbers with specified PDF.
% R = IMNOISE2(TYPE, M, N, A, B) generates an array, R, of size
% M-by-N, whose elements are random numbers of the specified TYPE
% with parameters A and B. If only TYPE is included in the
% input argument list, a single random number of the specified
% TYPE and default parameters shown below is generated. If only
% TYPE, M, and N are provided, the default parameters shown below
% are used. If M = N = 1, IMNOISE2 generates a single random
% number of the specified TYPE and parameters A and B.
%
% Valid values for TYPE and parameters A and B are:
%
% 'uniform'      Uniform random numbers in the interval (A, B).
%                 The default values are (0, 1).
% 'gaussian'     Gaussian random numbers with mean A and standard
%                 deviation B. The default values are A = 0, B = 1.
% 'salt & pepper' Salt and pepper numbers of amplitude 0 with
%                 probability Pa = A, and amplitude 1 with
%                 probability Pb = B. The default values are Pa =
%                 Pb = A = B = 0.05. Note that the noise has
%                 values 0 (with probability Pa = A) and 1 (with
%                 probability Pb = B), so scaling is necessary if
%                 values other than 0 and 1 are required. The noise
%                 matrix R is assigned three values. If R(x, y) =
%                 0, the noise at (x, y) is pepper (black). If
%                 R(x, y) = 1, the noise at (x, y) is salt
%                 (white). If R(x, y) = 0.5, there is no noise
%                 assigned to coordinates (x, y).
% 'lognormal'    Lognormal numbers with offset A and shape
%                 parameter B. The defaults are A = 1 and B =
%                 0.25.
% 'rayleigh'     Rayleigh noise with parameters A and B. The
%                 default values are A = 0 and B = 1.
% 'exponential' Exponential random numbers with parameter A. The
```

```

%           default is A = 1.
%   'erlang'   Erlang (gamma) random numbers with parameters A
%              and B. B must be a positive integer. The
%              defaults are A = 2 and B = 5. Erlang random
%              numbers are approximated as the sum of B
%              exponential random numbers.

%   Copyright 2002-2006 R. C. Gonzalez, R. E. Woods, & S. L. Eddins
%   Digital Image Processing Using MATLAB, Prentice-Hall, 2004
%   $Revision: 1.6 $   $Date: 2006/07/15 20:44:52 $

% Set default values.
if nargin == 1
    a = 0; b = 1;
    M = 1; N = 1;
elseif nargin == 3
    a = 0; b = 1;
end

% Begin processing. Use lower(type) to protect against input being
% capitalized.
switch lower(type)
case 'uniform'
    R = a + (b - a)*rand(M, N);
case 'gaussian'
    R = a + b*randn(M, N);
case 'salt & pepper'
    if nargin <= 3
        a = 0.05; b = 0.05;
    end
    % Check to make sure that Pa + Pb is not > 1.
    if (a + b) > 1
        error('The sum Pa + Pb must not exceed 1.')
    end
    R(1:M, 1:N) = 0.5;
    % Generate an M-by-N array of uniformly-distributed random numbers
    % in the range (0, 1). Then, Pa*(M*N) of them will have values <=
    % a. The coordinates of these points we call 0 (pepper
    % noise). Similarly, Pb*(M*N) points will have values in the range
    % > a & <= (a + b). These we call 1 (salt noise).
    X = rand(M, N);
    c = find(X <= a);
    R(c) = 0;
    u = a + b;
    c = find(X > a & X <= u);
    R(c) = 1;
case 'lognormal'
    if nargin <= 3
        a = 1; b = 0.25;
    end
    R = exp(b*randn(M, N) + a);
case 'rayleigh'
    R = a + (-b*log(1 - rand(M, N))).^0.5;
case 'exponential'
    if nargin <= 3
        a = 1;
    end
    if a <= 0
        error('Parameter a must be positive for exponential type.')
    end
    k = -1/a;
end

```

```

    R = k*log(1 - rand(M, N));
case 'erlang'
    if nargin <= 3
        a = 2; b = 5;
    end
    if (b ~= round(b) | b <= 0)
        error('Param b must be a positive integer for Erlang.')
    end
    k = -1/a;
    R = zeros(M, N);
    for j = 1:b
        R = R + k*log(1 - rand(M, N));
    end
otherwise
    error('Unknown distribution type.')
end

```

Порядок выполнения работы

1. Синтезировать пустое полутоновое 8-битное изображение размером 800×800 пикселей.
2. Нанести на изображение шум в соответствии с вариантом индивидуального задания по известному закону распределения.
3. Построить гистограмму распределения полученного изображения шума. Сохранить график гистограммы в директорию «...DIP\Lab2».
4. Синтезировать изображение Объекта 1 в соответствии с вариантом индивидуального задания в центре изображения (см. Рисунок 2). Сохранить изображение в директорию «...DIP\Lab2».
5. Выполнить масштабирование изображения, увеличив и уменьшив размер синтезированного объекта в 2 раза разными методами (ближайшего соседа, билинейной или бикубической интерполяции). Сохранить изображения в директорию «...DIP\Lab2».
6. Синтезировать новое изображение с шумом и нанести объекты 2 и 3 в соответствии с вариантом индивидуального задания. Объекты должны располагаться в верхнем левом и нижнем правом углах синтезированного изображения (см. Рисунок 2). Размеры объектов подобрать самостоятельно, исходя из того, что наибольшая сторона (или диаметр) должна быть не менее 80 пикселей.
7. Выполнить операцию зеркального отражения по горизонтали. Сохранить изображение в директорию «...DIP\Lab2».
8. Выполнить операцию зеркального отражения по вертикали. Сохранить изображение в директорию «...DIP\Lab2».
9. Выполнить операцию поворота изображения по часовой стрелке на 45°. Сохранить изображение в директорию «...DIP\Lab2».
10. Выполнить операцию поворота изображения против часовой стрелки на 45°. Сохранить изображение в директорию «...DIP\Lab2»

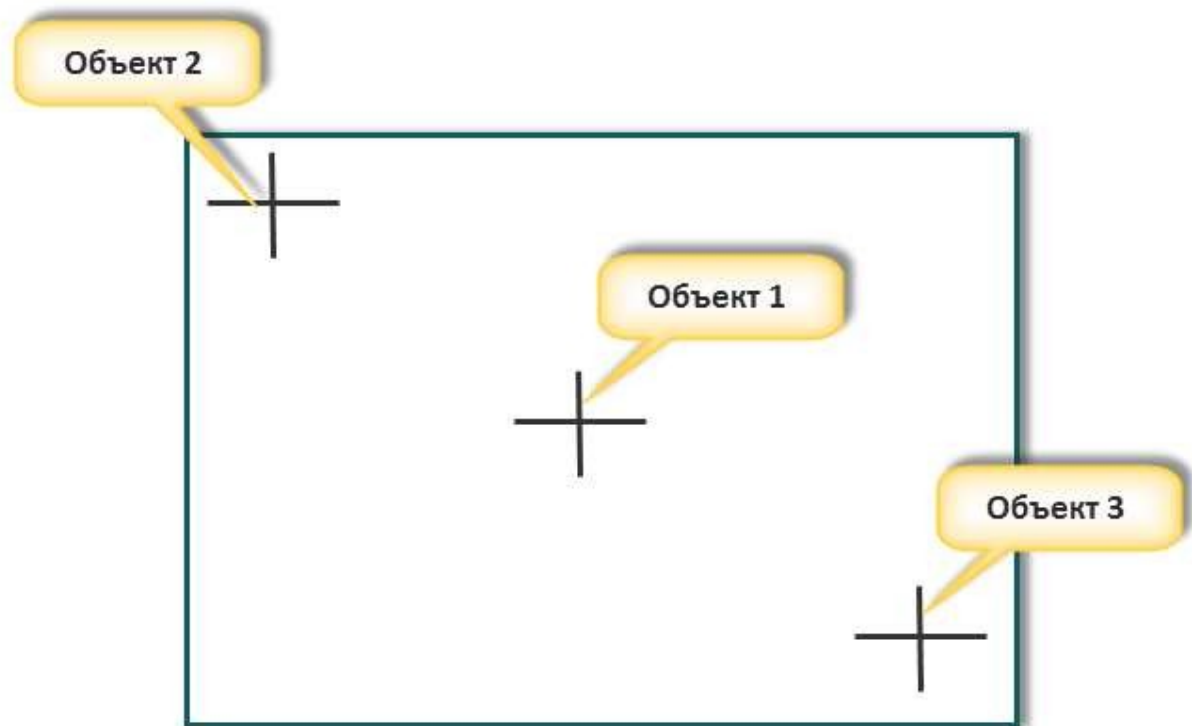




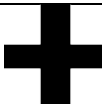










































Рисунок 2 – Расположение синтезированных объектов на изображении

11. Выбрать из папки вспомогательных материалов «...\Labs\Lab2\Fon» любое из предложенных изображений фона.
12. Вырезать из изображения фона любой участок размером 800×800 пикселей.
13. Уменьшить яркость вырезанного фона в 4 раза.
14. Сформировать новое полутоновое изображение из вырезанного фона с уменьшенной яркостью, 2-х объектов и добавить на изображение шум в соответствии с вариантом индивидуального задания. Сохранить полученное изображение в директорию «...DIP\Lab2».
15. Выполнить операцию «Негатив» для синтезированного изображения. Сохранить полученное изображение в директорию «...DIP\Lab2».
16. Сформировать новое полутоновое изображение из вырезанного фона с уменьшенной яркостью, одного объекта и добавить на изображение шум в соответствии с вариантом индивидуального задания. Сохранить полученное изображение в директорию «...DIP\Lab2».
17. Выполнить операцию разности двух изображений, полученных в пп. 14 и 16. Сохранить полученное изображение в директорию «...DIP\Lab2».

Варианты индивидуального задания

Примечание. Синтезированный объект должен быть белого цвета. Размеры объекта выбираются таким образом, чтобы он был различим на изображении.

| № | Объект 1 | Объект 2 | Объект 3 | Шум |
|-----|---|---|---|---------------------------|
| 1. |  |  |  | Рэлея |
| 2. |  |  |  | «Соль и перец» |
| 3. |  |  |  | Экспоненциальный |
| 4. |  |  |  | Эрланга |
| 5. |  |  |  | Логарифмически нормальный |
| 6. |  |  |  | Равномерный |
| 7. |  |  |  | Гауссов |
| 8. |  |  |  | «Соль и перец» |
| 9. |  |  |  | Гауссов |
| 10. |  |  |  | Экспоненциальный |

| | | | | |
|-----|---|---|--|------------------|
| 11. |  |  |  | «Соль и перец» |
| 12. |  |  |  | Рэля |
| 13. |  |  |  | Равномерный |
| 14. |  |  |  | Эрланга |
| 15. |  |  |  | Экспоненциальный |

Вопросы для подготовки к защите

1. Простейшие геометрические преобразования изображения.
2. Аффинные преобразования изображения.
3. Простейшие операции над изображением (арифметические операции). Линейные и нелинейные операции.
4. Шумы на изображении. Виды шума и источники шума на изображении.
5. Гистограмма изображения и ее свойства.