МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра прикладної математики

Звіт із проходження переддипломної практики

за темою:

«МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОЗПІЗНАВАННЯ ТЕКСТУРНИХ ЗОБРАЖЕНЬ ЗА ДОПОМОГОЮ ФРАКТАЛЬНОГО АНАЛІЗУ»

Виконав: СТАНКЕВИЧ М. П.

Група: КМ-21

Керівник: ТЕРЕЩЕНКО І. О.

Київ – 2016

АНОТАЦІЯ ЗВІТУ ПРО ПЕРЕДДИПЛОМНУ ПРАКТИКУ

Даний звіт присвячено результатам проходження переддипломної практики на тему: «Математичне та програмне забезпечення розпізнавання текстурних зображень за допомогою фрактального аналізу», яка відбувалася з 11 квітня по 1 травня 2016 року на базі інституту Економіки та Прогнозування НАН України.

У рамках практики було поставлено задачу на дипломне проектування, сформульовано критерії вибору методу розв’язання поставленої задачі. Були розглянуті методи сегментації текстурних зображень, коротко сформовано суть кожного з них та вказано доцільність їх використання. Розглядалися такі методи сегментації як: статистичний, структурний, фрактальний. Поміж фрактальних методів сегментації були виділені та детально розглянуті методи, що використовують розмірності Хаусдорфа та Мінковського. Також було розглянуто допоміжний метод визначення порогу бінарізації – метод Оцу, а також Наївний Баєсів класифікатор. Було порівняно різні методи сегментації зображень та визначено той, що відповідає якнайкраще потребам поставленої задачі.

На основі особливостей застосування розроблюваних методів розпізнавання сегментованої текстури було обрано для реалізації метод фрактальної сегментації з використанням розмірності Мінковського.

АННОТАЦИЯ ОТЧЕТА О ПРЕДДИПЛОМНОЙ ПРАКТИКЕ

Данный отчет посвящен результатам прохождения преддипломной практики на тему: «Математическое и программное обеспечение распознавания текстурных изображений с помощью фрактального анализа», которая проходила с 11 апреля по 1 мая 2016 на базе института Экономики и Прогнозирования НАН Украины.

В рамках практики была поставлена ​​задача на дипломное проектирование, сформулированы критерии выбора метода решения поставленной задачи. Были рассмотрены методы сегментации текстурных изображений, коротко сформирован суть каждого из них и указано целесообразность их использования. Рассматривались такие методы сегментации как: статистический, структурный, фрактальный. Между фрактальных методов сегментации были выделены и подробно рассмотрены методы, использующие размерности Хаусдорфа и Минковского. Также были рассмотрены вспомогательный метод определения порога бинаризации - метод Оцу, а также Наивный Байеса классификатор. Было сравнительно различные методы сегментации изображений и определены отвечающий лучше потребностям поставленной задачи.

На основе особенностей применения разрабатываемых методов распознавания сегментированной текстуры были выбраны для реализации метод фрактальной сегментации с использованием размерности Минковского.

ANNOTATION ABOUT PRE-DIPLOMA PRACTICE REPORT

This report is devoted to the results of the Pre-diploma Practice titled: «Mathematical framework and program software for texture images recognition using fractal analysis» that took place from April, 11 to May, 1, 2016 at the Institute of Economics and Forecasting of NAS of Ukraine.

During the practice, the task for the tesis was set, criteria for choosing the method for solving this task are formulated. There were considered methods of segmentation of texture images briefly formed the essence of each of them and indicated feasibility of their use. There were considered methods such as segmentation, statistical, structural, fractal. Among fractal segmentation methods have been identified and examined in detail the methods used Hausdorff dimension and Minkowski. It was also considered a secondary method of determining the threshold binarization Otsu method, and the naive Bayesian classifier. There were compared different methods of image segmentation and defined the one that best meets the needs of the task.

Based on the features of future application of texture image recognition method was chosen to implement segmentation method using fractal Minkowski dimension.

ЗМІСТ

[ВСТУП 7](#_Toc451137444)

[1 ПОСТАНОВКА ЗАДАЧІ 8](#_Toc451137445)

[2 ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ 9](#_Toc451137446)

[2.1 Реалізація сегментації текстурних зображень засобами MATLAB Іmage Processing Toolbox 9](#_Toc451137447)

[2.2 Програмні застосунки доповненої реальності 15](#_Toc451137448)

[2.2.1 Wikitude Drive 15](#_Toc451137449)

[2.2.2 Route 66 Maps + Navigation 17](#_Toc451137450)

[2.2.3 Transparent Bonnet 17](#_Toc451137451)

[2.3 Проблеми існуючих реалізацій 19](#_Toc451137452)

[2.4 Висновки 20](#_Toc451137453)

[3 ОГЛЯД МАТЕМАТИЧНИХ МЕТОДІВ 21](#_Toc451137454)

[3.1 Класифікація методів сегментації текстур 21](#_Toc451137455)

[3.1.1 Статистичні методи аналізу текстурних областей зображень 21](#_Toc451137456)

[3.1.2 Структурні методи аналізу текстурних областей зображень. 22](#_Toc451137457)

[3.1.3 Фрактальні методи аналізу текстурних областей зображень. 23](#_Toc451137458)

[3.1.4 Спектральні методи аналізу текстурних областей зображень. 25](#_Toc451137459)

[3.1.5 Комбіновані методи аналізу текстурних областей зображень. 25](#_Toc451137460)

[3.2 Статистичний підхід до сегментації текстур 26](#_Toc451137461)

[3.3 Фрактальний підхід до сегментації текстур 27](#_Toc451137462)

[3.3.1 Розмірність Хаусдорфа 27](#_Toc451137463)

[3.3.2 Фрактальна розмірність 28](#_Toc451137464)

[3.3.3 Розмірність Мінковського 30](#_Toc451137465)

[3.3.4 Box-counting алгоритм. 31](#_Toc451137466)

[3.4 Метод Оцу обчислення порога бінарізації 32](#_Toc451137467)

[3.5 Наївний Баєсів класифікатор 34](#_Toc451137468)

[3.6 Висновки 36](#_Toc451137469)

[ПЕРЕЛІК ПОСИЛАНЬ 37](#_Toc451137470)

# ВСТУП

Колір і текстура є важливими характеристиками зображення. Проблема аналізу кольорових текстур включає в себе такі аспекти, як опис кольорових текстур, їх класифікація, тобто формування кластерів (при цьому під кластером розуміють зазвичай групу об'єктів, що утворюють в просторі опису компактну в деякому сенсі область), і сегментація, тобто розбиття зображення на області, які є однорідними щодо однієї або декількох характеристик, або належать деякому кластеру. Класично виділяють два підходи до опису текстур: статистичний і структурний. Сьогодні можна сказати, що розвивається і набув поширення третій підхід до опису текстур - фрактальний. Такий підхід є дуже ефективним в умовах аналізу природних зображень, на відміну від класичних методів.

Аналіз та розпізнавання за текстурою зображень є важливою основою для побудування програмних застосунків доповненої реальності. Такі за стосунки на разі є актуальною галуззю, що динамічно розвивається та знаходить застосування у багатьох сферах діяльності – від побутових потреб та освітніх програм до реалізації важливих промислових потреб.

Буде розглядатись застосунок такого аналізу на прикладі програми автомобільного навігатору з функцією доповненої реальності.

# 1 ПОСТАНОВКА ЗАДАЧІ

Метою даної роботи є створення математичного та програмного забезпечення для розпізнавання текстурних зображень за допомогою фрактального методу сегментації. У якості предметної області використовується відео потік зображень дороги, що подається з відео-регістратора автомобіля.

Основні задачі, що підлягають розв‘язку:

1. проведення порівняльного аналізу існуючих методів сегментації текстурних зображень;
2. вибір методу для розв’язання поставленої задачі;
3. розробка програмного забезпечення розпізнавання текстури;
4. тестування розробленої системи на прикладі зображень відео регістратора автомобіля.

Реалізована система має задовольняти такі вимоги:

1. обраний алгоритм має виділяти межі області дороги на поданому зображенні з похибкою не більше 5% від розміру, що вона займає на вхідному зображенні;
2. програма має оброблювати потік зображення, що подається з відеокамери регістратора з затримкою не більше 1.5 с;
3. система має адекватно працювати при зміні якості зображення, що оброблюється;
4. реалізація обраного алгоритму має не втрачати свої властивості при зміні області.

# 2 ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ

На даний момент більшість розробок з розпізнавання текстурних зображень та за стосунку цих методів для добудови доповненої реальності знаходяться у стадії теоретичних досліджень та тестових проектів. Дана сфера молодою та інноваційною, тому існує багато пілотних рішень, що на даний момент конкурують між собою, але ще немає однозначних лідерів, ідеальних реалізацій поставлених задач.

Цьому розділі буде розглянуто реалізації стандартних методів сегментації текстурних зображень, що реалізує більшість математичних пакетів.

Також буде розглянуто програмні продукти з побудови доповненої реальності для відео регістраторів автомобілів.

## 2.1 Реалізація сегментації текстурних зображень засобами MATLAB Іmage Processing Toolbox

Розглянемо реалізацію статистичного підходу до сегментації текстурних зображень на прикладі використання текстурних фільтрів. Такі фільтри вже реалізовані в розширенні Іmage Processing Toolbox пакету MATLAB.

Для візуалізації нашого прикладі оберемо задачу сегментації двох видів тканини на зображенні з використанням текстур фільтрів.

Алгоритм:

* Крок 1: Зчитування зображення.
* Крок 2: Виділення текстури зображення.
* Крок 3: Створення грубої маски для виділення нижньої текстури.
* Крок 4: Використання грубої маски для сегментації верхньої текстури.
* Крок 5: Відображення результатів сегментації.
* Використання інших текстурних фільтрів для сегментації.

*Крок 1: Зчитування зображення.*

Зчитуємо зображення з файлу bag.png, що наданий в пакеті MATLAB у якості типового прикладу зображення,що містить яскраво виражені текстури (рис. 6.1.1).

I = imread ( 'bag.png');

figure, imshow (I);

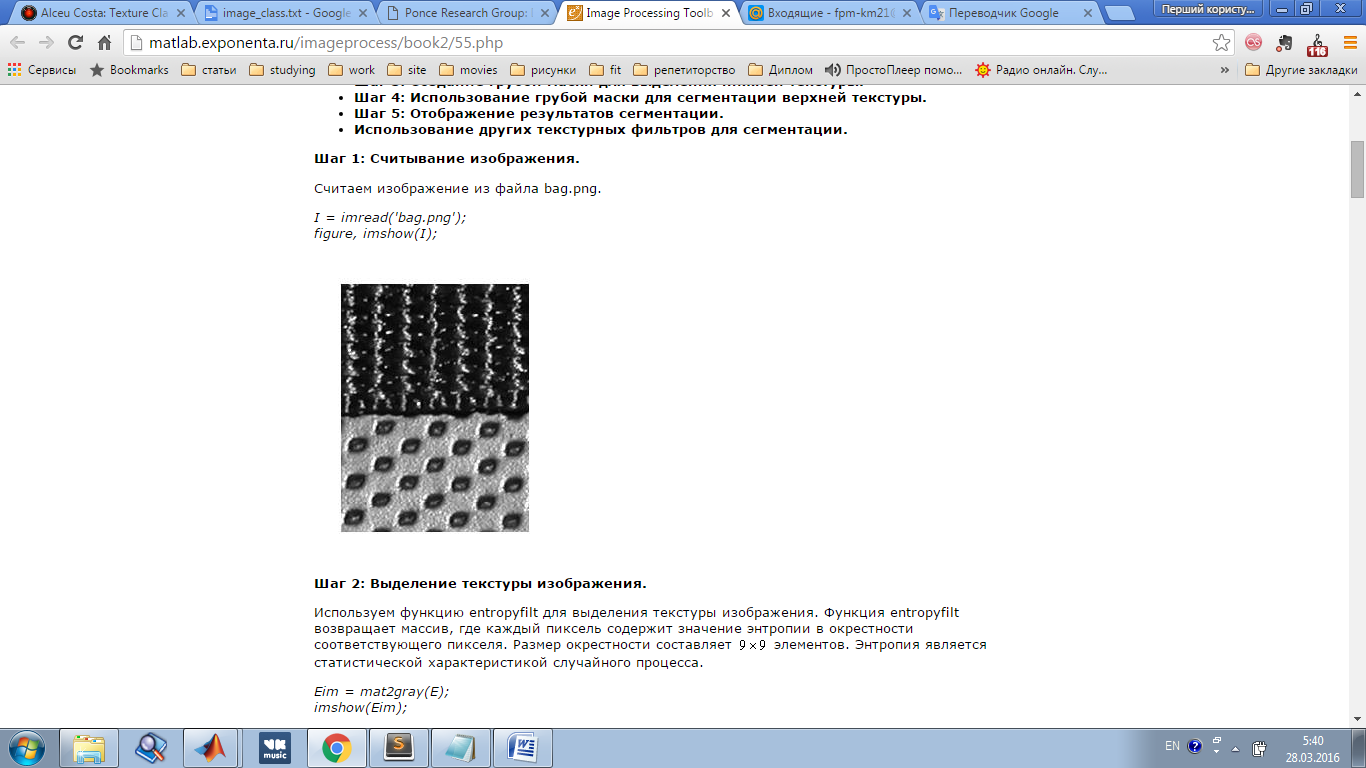


Рис. 2.1.1 Вихідне зображення bag.png

*Крок 2: Виділення текстури зображення.*

Використовуємо функцію entropyfilt для виділення текстури зображення. Функція entropyfilt повертає масив, де кожен піксель містить значення ентропії в околиці відповідного пікселя. Розмір околиці становить 9\*9 елементів. Ентропія є статистичною характеристикою випадкового процесу.

E=entropyfilt(I);

Eim = mat2gray (E);

imshow (Eim);

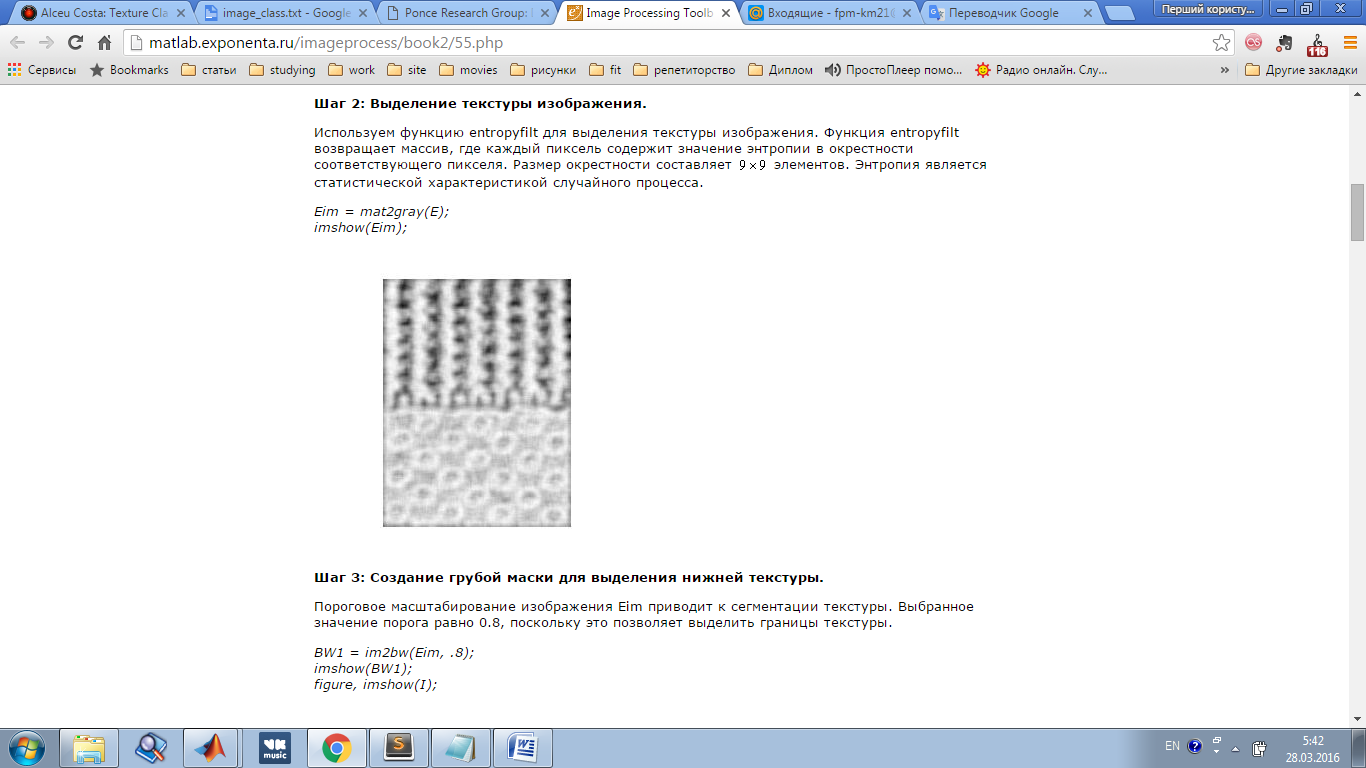


Рис. 2.1.2 Зображення після обробки функцією entropyfilt

*Крок 3: Створення грубої маски для виділення нижньої текстури.*

Граничне масштабування зображення Eim призводить до сегментації текстури. Вибране значення порога одно 0.8, оскільки це дозволяє виділити кордону текстури.

BW1 = im2bw (Eim, .8);

imshow (BW1);

figure, imshow (I);



Рис. 2.1.3 Результати бінарної сегментаціїі

Сегментовані об'єкти представлені на бінарному зображенні BW1(рис. 6.1.3). При порівнянні зображень BW1 і I відзначимо, що верхня текстура сегментована занадто надмірно (багато білих об'єктів), а нижня текстура сегментована майже нероздільні. Для виділення нижньої текстури можна використовувати також функцію bwareaopen.

BWao = bwareaopen (BW1,2000);

imshow (BWao);

Для згладжування меж можна використовувати функцію imclose, а також морфологічні операції відкриття і закриття об'єктів на зображенні BWao. При обробці використовуються ті ж околиці, що і в функції entropyfilt.

nhood = true (9);

closeBWao = imclose (BWao, nhood);

imshow (closeBWao)

Використовується функція imfill для заповнення об'єктів в closeBWao.

roughMask = imfill (closeBWao, 'holes');



Рис. 2.1.4 Бінарізація для виділення нижньої текстури

*Крок 4: Використання грубої маски для сегментації верхньої текстури.*

Порівняємо бінарне roughMask і вихідне зображення I (рис.6.1.5). Відзначимо, що маска для нижньої текстури не є найкращою. Тому будемо використовувати roughMask для сегментації верхньої текстури.



Рис. 2.1.5 Бінарізація за допомогою грубої маски

imshow (roughMask);

figure, imshow (I);

Отримаємо таке зображення верхньої текстури з використанням roughMask.

I2 = I;

I2 (roughMask) = 0;

imshow (I2);

Використоовуємо функцію entropyfilt для обчислення текстури зображення.

E2 = entropyfilt (I2);

E2im = mat2gray (E2);

imshow (E2im);

Поріг E2im розраховується з використанням graythresh.

BW2 = im2bw (E2im, graythresh (E2im));

imshow (BW2)

figure, imshow (I);

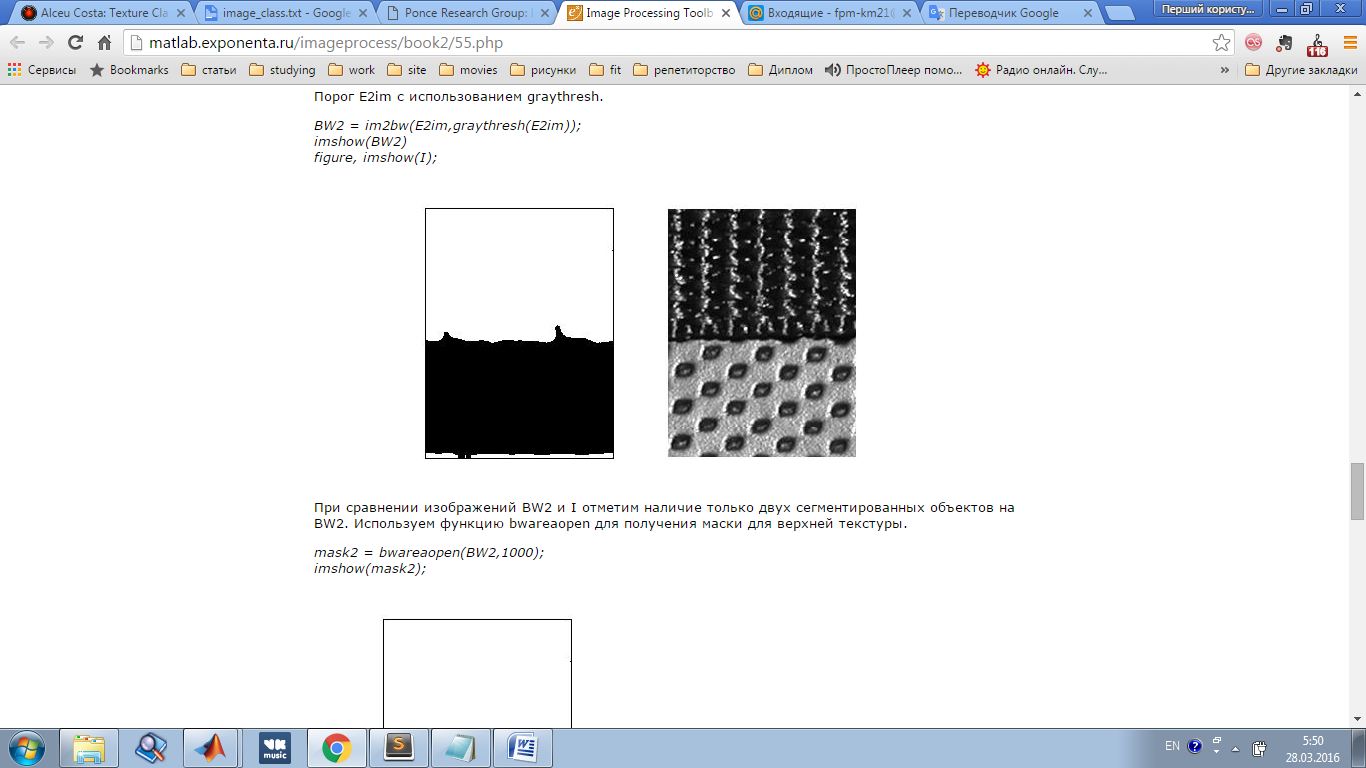


Рис. 2.1.6 Бінарізація з додатковим розрахунком порогу

При порівнянні зображень BW2 і I відзначимо наявність тільки двох сегментованих об'єктів на BW2. Використовуємо функцію bwareaopen для отримання маски для верхньої текстури.

mask2 = bwareaopen (BW2,1000);

imshow (mask2);

*Крок 5: Відображення результатів сегментації.*

Використовуємо mask2 для отримання верхньої і нижньої текстури зображення I.

texture1 = I;

texture1 (~ mask2) = 0;

texture2 = I;

texture2 (mask2) = 0;

imshow (texture1);

figure, imshow (texture2);

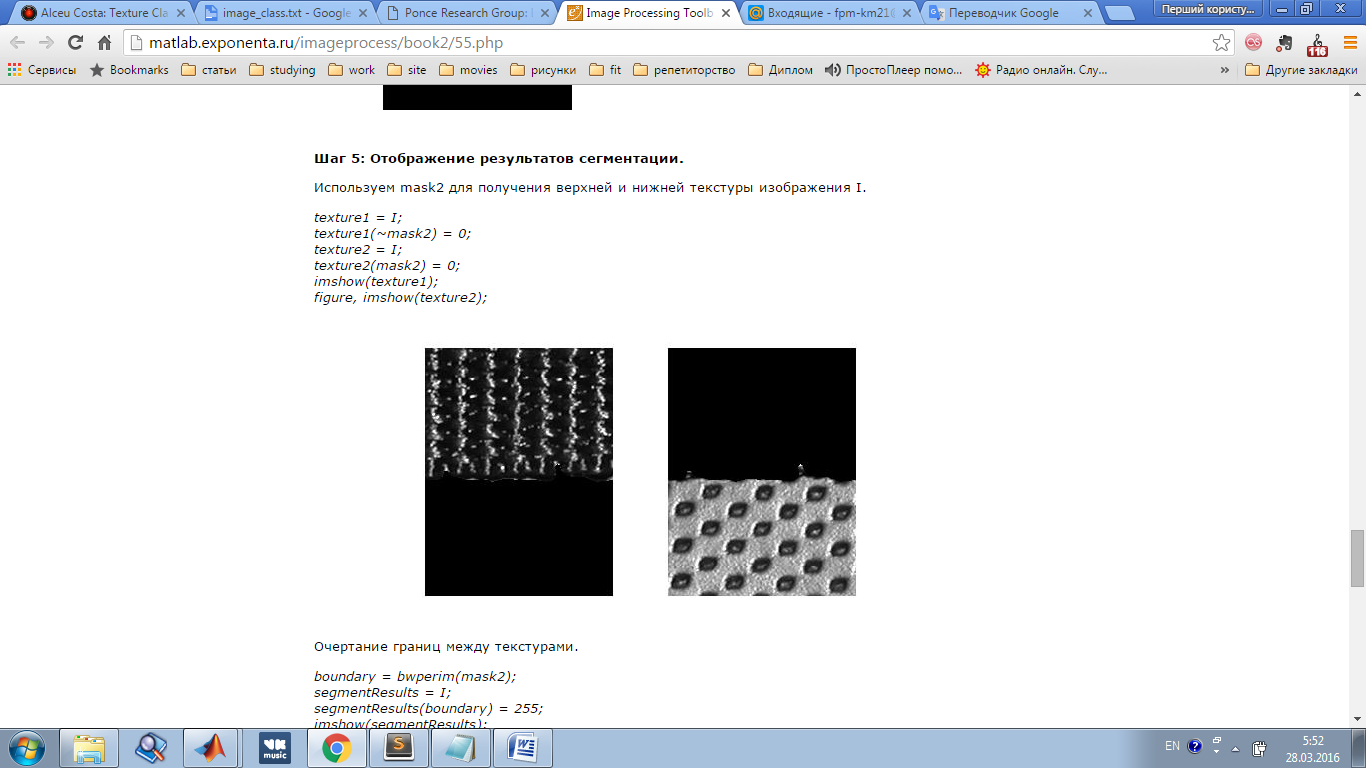


Рис. 2.1.7 Накладання масок для верхньої та нижньої текстур

Обрис кордонів між текстурами знайдемо наступним чином:

boundary = bwperim (mask2);

segmentResults = I;

segmentResults (boundary) = 255;

imshow (segmentResults);



Рис.2.1.8 Обрис кордону між текстурами

*Використання інших текстурних фільтрів для сегментації.*

Замість функції entropyfilt можна використовувати функції stdfilt і rangefilt разом з іншими морфологічними функціями для отримання аналогічних результатів сегментації.

S = stdfilt (I, nhood);

imshow (mat2gray (S));

R = rangefilt (I, ones (5));

imshow (R);

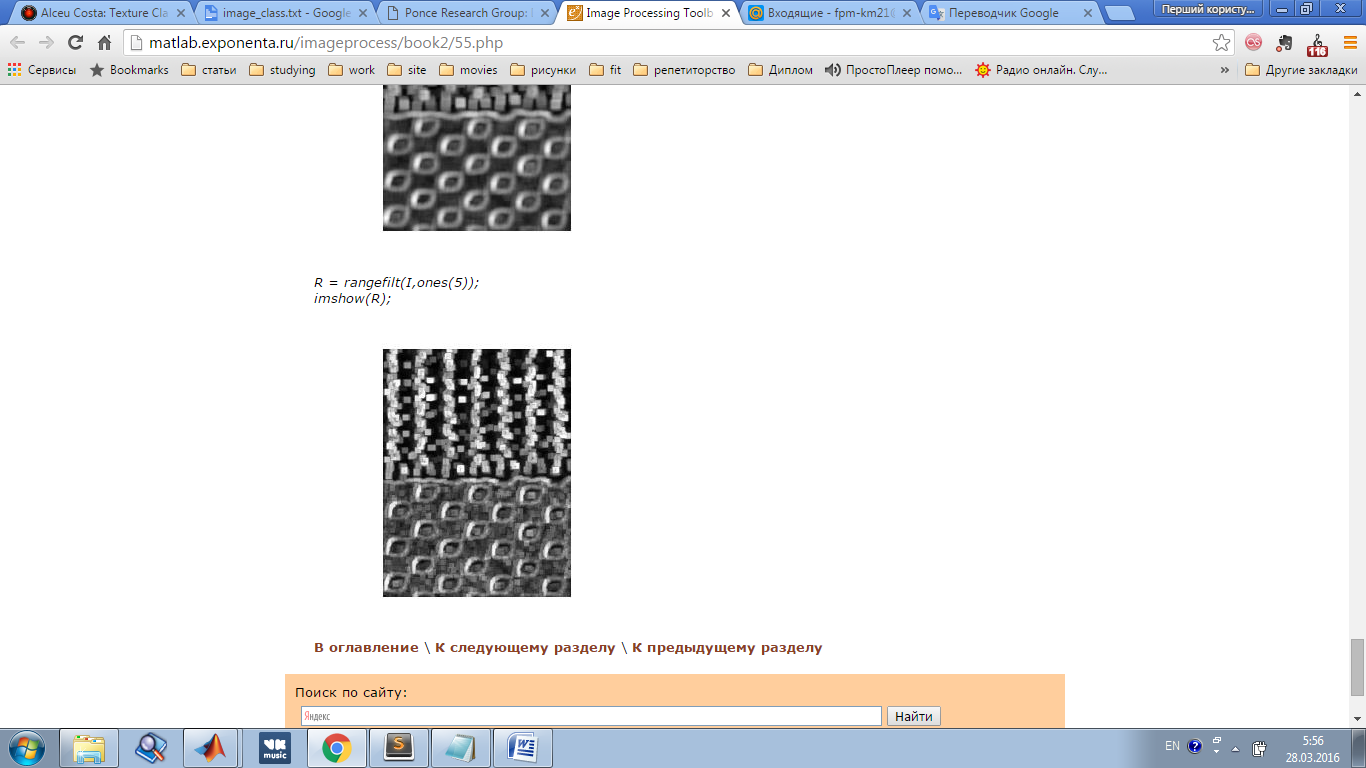
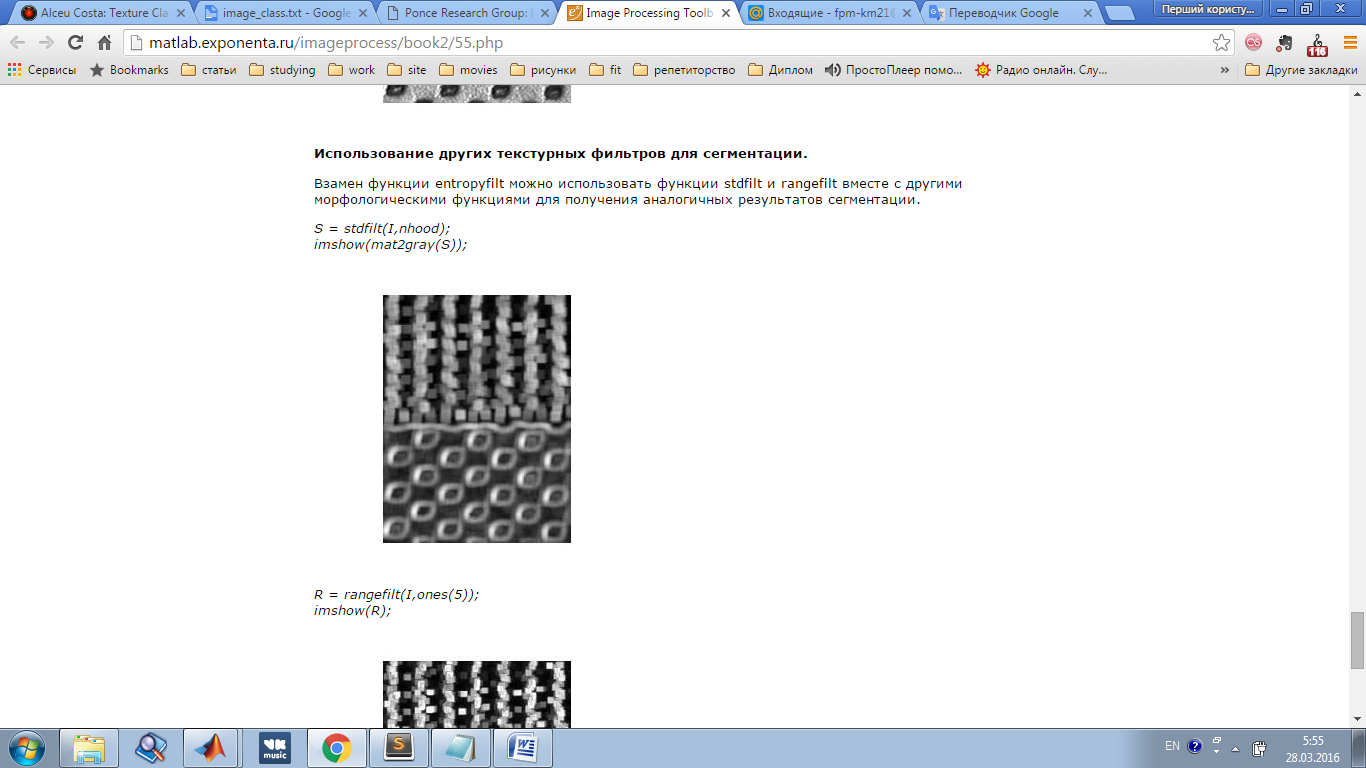
****

Рис. 2.1.9 Сегментація за допомогою середньоквадратичного та рангового фільтрів

## 2.2 Програмні застосунки доповненої реальності

### 2.2.1 Wikitude Drive

У Німеччині компанія Wikitude представила навігаційне програмне забезпечення, що використовує доповнену реальність - Wikitude Drive. Додаток призначений для комунікаторів, що працюють під управлінням операційної системи Android 2.1 і вище і на даний момент працює в Австрії, Німеччині та Швейцарії, але пізніше вона почне працювати і в інших країнах.

Основна відмінність Wikitude Drive полягає в тому, що програма накладає маршрут на реальне зображення, що отримується за допомогою відеокамери комунікатора - на думку компанії, такий спосіб краще допомагає водієві орієнтуватися на невідомій місцевості.

Ще одна особливість Wikitude Drive полягає в тому, що всі інші навігаційні системи час від часу змушують водія відірвати очі від дороги і подивитися абстрактну навігаційну карту, але якщо просто поглянути на екран карти протягом однієї секунди при русі на швидкості 100 км / год, то водій фактично стає "сліпим" на 28 метрів, а для розуміння того, що намальовано на екрані, однієї секунди може не вистачити. Така ситуація відволікання під час керування автомобілем може призвести навіть до аварій. З Wikitude Drive все трохи простіше - отримання певних у верхній частині екрану, а на основній частині реальне зображення того, що ви бачите перед собою (дивитися теж треба, але вникати потрібно менше).

Звичайно ж, у такій навігації є і недоліки - їй неможливо користуватися в темний час доби.

Картографічна основа програми надана одним зі світових постачальників – компанією Navteq, а в наступних версіях в Wikitude Drive Wikitude з'явиться World Browser, який надасть інформацію про мільйони точок POI і доступ в мережу до призначеного для користувача контенту.



Рис. 2.2.1.1 Робота програми Wikitude Drive

### 2.2.2 Route 66 Maps + Navigation

Компанія Route 66 при випуску програми «Route 66 Maps + Navigation» вирішила застосувати творчий підхід до GPS-навігації, зробивши можливим використання додатка усіма власниками смартфонів і планшетів на базі Android.

Особливістю цієї програми є наявність доповненої реальності. Тепер, якщо автомобіліст знаходиться в дорозі, крім маршруту до пункту призначення, смартфон показує і віртуальний автомобіль з рекомендаціями для водія за вибором шляху поїздки на багатьох мовах. Сам процес відбувається за допомогою використання камери, де габаритні вогні автомобіля і стрілка вказують напрямок руху.



Рис. 2.2.2.1 Route 66 Maps + Navigation

### 2.2.3 Transparent Bonnet

компанія Land Rover опублікувала на своєму офіційному сайті тизер-відео і кілька рендерів «прозорого» капота автомобіля з підтримкою технології доповненої реальності. Дане рішення здатне надати водієві цифрову проекцію поверхні під колесами автомобіля, що в свою чергу дозволить суттєво розширити поле зору водія.

Система отримала назву Transparent Bonnet. Вона представлена ​​кількома камерами, розташованими в області решітки радіатора автомобіля, і технологією Head-Up Display, яка передбачає проектування повнокольорового зображення прямо на лобове скло. Саме за рахунок камер і технології Head-Up Display забезпечується віртуальна «прозорість» капота. Водій при цьому отримує можливість бачити поверхню дороги, яка зазвичай через капота і особливостей конструкції автомобіля надає поза увагою.

Принцип роботи Transparent Bonnet в загальних рисах досить простий: після активації камери починають записувати відео, а технологія Head-Up Display дозволяє виводити його в нижній частині лобового скла в реальному часі, створюючи таким чином враження прозорості капота і двигуна.

Система Прозорий Bonnet поки знаходиться на ранній стадії розробки. Однак вже зараз фахівці Land Rover впевнені, що подібна система може стати незамінним помічником водія при здійсненні руху вгору по похилій поверхні або по нерівній місцевості. Технологія також зможе показати свою цінність при здійсненні маневрування в обмеженому просторі.



Рис. 2.2.3.1 Transparent Bonnet

## 2.3 Проблеми існуючих реалізацій

Технології доповненої реальності впроваджуються в автомобілі і можуть бути дуже корисними для водія. Однак користь вони приносять тільки тоді, коли доопрацьовані і продумані.

Вчені виявили, що подача інформації у нинішніх технологій доповненої реальності для автомобіля, зовсім неправильна. AR-девайси для автомобіліста відволікають увагу водія і можуть спровокувати ДТП. Це підтверджує фахівець Іан Спенс з Університету Торонто: "Водій автомобіля повинен бути зосереджений на дорозі, а не на значках, вискакують на лобовому склі машини".

Іан Спенс провів експеримент, щоб підтвердити свої слова. Учасники виконували пару завдань на комп'ютері, де треба було порахувати точки, які хаотично розміщені на екрані монітора. Підрахунок треба було робити максимально швидко.

До того ж до підрахунку точок, іноді з'являвся чорний прямокутник, які відволікав учасників експерименту. Точки і прямокутник могли з'явитися разом, а їх форма могла змінюватися.

Учасники називали точну кількість точок, коли прямокутника не було. Але коли він з'являвся, більшість піддослідних або не відзначала його, або помилялася в кількості точок.

Через те, що учасники намагалися виконати відразу два завдання в один момент, їх увагу розсіюється, через це жодна із завдань не могла бути виконана вірно. Цей експеримент підтверджує, що доповнена реальність для автолюбителя поки що зайве. Можливо, в найближчому майбутньому, коли інформація буде мати правильну подачу і розміщення, доповнена реальність буде забезпечувати комфортне і просте переміщення по міських дорогах.

## 2.4 Висновки

На даний момент існує ряд програм, що застосовують технології доповненої реальності для автомобільних навігаторів. Однак більшість з них має недоліки, головний з яких – точність та швидкість реакції. Такі програми можуть погано працювати в темний час доби, а також зазвичай потребують підключення до великих масивів карт.

Труднощі в цій сфері як було зазначено в минулому розділі є суттєвими, бо системи, що допомагають в керуванні автомобілем є безпеко критичними. Запропонований фрактальний метод аналізу зображень може вирішити існуючі проблеми.

Природа фрактального методу дасть змогу працювати з зображеннями, що отримуються навіть у сутінках, що є важливим моментом, бо розширяє застосованіт подібних програм майже в два рази.

Також в даній роботі будемо обробляти інформацію щодо дороги в онлайн режимі, зосереджуючись на характері поверхні попереду машини, а не даних, отриманих з масиву карт. Таких підхід підвищує швидкість реагування програми, а також зменшує об’єм необхідної для неї завантаженої інформації, дозволяє використовувати її на новій, не відомій заздалегідь місцевості, без зв’язку з мережею інтернет.

Отож, використання фрактального підходу розпізнавання текстурних зображень в даній предметній області надасть змогу отримати не тільки приємний візуально та зручний для водія інструмент, а і зробить процес водіння більш безпечним.

# 3 ОГЛЯД МАТЕМАТИЧНИХ МЕТОДІВ

## 3.1 Класифікація методів сегментації текстур

Існує кілька основних підходів до класифікації методів сегментації текстур. Методи текстурной сегментації розробляються на основі двох основних підходів: аналіз областей і аналіз меж. Методи на основі областей намагаються знайти групи або кластери пікселів зі схожими текстурними властивостями. Методи на основі кордонів намагаються виявити границі текстур між пікселями, яким відповідають різні текстурні розподіли. Залежно від використовуваних ознак текстурних областей зображень, методи сегментації на основі аналізу областей можна розділити на статистичні, структурні, фрактальні, спектральні і комбіновані методи.

### 3.1.1 Статистичні методи аналізу текстурних областей зображень

При статистичному підході до аналізу текстурних областей зображення трактуються як реалізації деякого випадкового процесу. Статистичні методи аналізу текстур засновані на обчисленні порядкових статистичних ознак текстурних областей зображення. Існують методи розпізнавання з використанням таких статистичних характеристик як автокореляційні функції, щільність перепадів і довжин серій. Статистичний текстурний аналіз заснований на обчисленні матриць суміжності, які враховують як рівні яскравості віліків, так і відносне розташування відліків з певною яскравістю на зображенні. Відлік двовимірної матриці суміжності для зображення показують оцінку ймовірності спільного появи на зображенні на відстані один від одного, що задається певним вектором зміщення відліків, з відповідними значеннями рівнів яскравості. Аналогічно, відліки тривимірної матриці показують оцінку ймовірності спільного появи на зображенні на заданому відстані один від одного трійки отсчѐтов з відповідними значеннями рівнів яскравості. З використанням матриць суміжності обчислюються текстурні ознаки, такі як кутовий момент, кореляція, контраст, інерція, ентропія, затінення, і ін. Матриця суміжності забезпечує інваріантність ознак до повороту, зсуву і масштабування. Ознаки, обчислені на основі двовимірної матриці - 2D ознаки, їх тривимірні аналоги - 3D ознаки. Таким чином, для кожного зображення можна обчислити вектор ознак - статистичну сигнатуру текстури.

Недоліками методів сегментації за статистичними ознаками є:

- висока детальність сегментації, що призводить до необхідності використання алгоритмів зменшення детальності; задача зниження детальності сегментації полягає в зниженні числа вихідних сегментів шляхом їх злиття. В результаті на сегментованому зображенні має залишитися невелике число великих сегментів, розділених добре помітними межами;

- висока обчислювальна складність;

- велике розмаїття статистичних ознак текстур викликає необхідність попереднього дослідження найбільш інформативних ознак в залежності від типу текстури.

### 3.1.2 Структурні методи аналізу текстурних областей зображень.

Структурні методи аналізу текстурних областей зображень засновані на тому, що текстура складається з регулярно або майже регулярно повторюється сукупності добре поділюваних примітивів (Мікротекстура), які розташовані згідно деякому правилу розміщення і ієрархічно об'єднуються в просторові впорядковані структури (макротекстури). Для опису текстури структурними методами необхідно визначити примітиви і правила їх об'єднання. Структурні методи добре підходять для аналізу регулярних текстур, що складаються з простих регулярних примітивів. Як тільки елементи текстури ідентифіковані, можливе застосування двох основних підходів до аналізу текстури. При першому підході обчислюють статистичні ознаки витягнутих елементів текстури і використовують їх як елементарні текстурні ознаки. При другому підході необхідно виділити принцип розташування примітивів, який описує текстуру. Останній підхід може включати в себе геометричний або синтаксичні методи аналізу текстури, наприклад аналіз діаграм полігонів Вороного. Метод аналізу зазвичай залежить від геометричних властивостей текстурних елементів. Структурний підхід застосується для регулярних штучних образів.

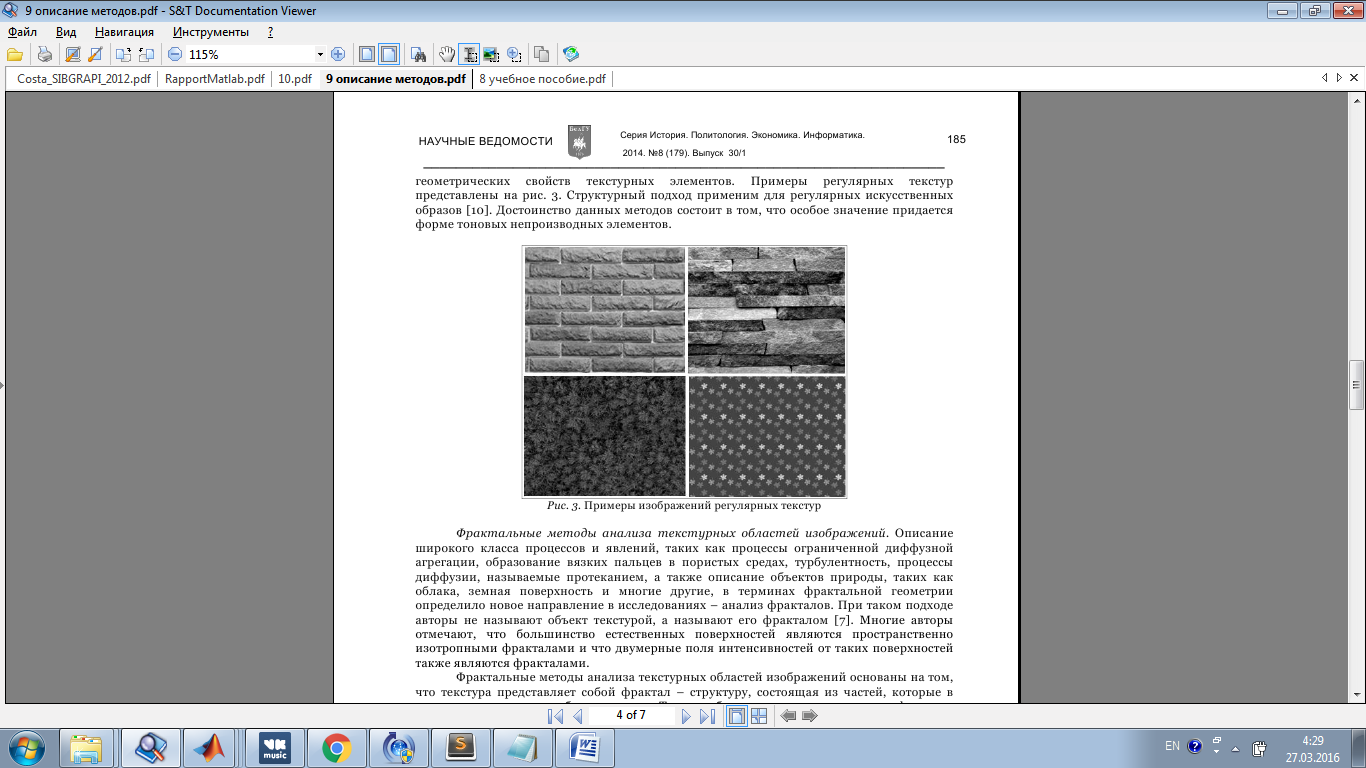


Рис. 3.1.2.1 Приклади зображень регулярних текстур

### 3.1.3 Фрактальні методи аналізу текстурних областей зображень.

При такому підході об'єкт не називають текстурою, а називають його фракталом. Багато авторів відзначають, що більшість природних поверхонь є просторово ізотропним фракталами і що двовимірні поля інтенсивностей від таких поверхонь також є фракталами.

Фрактальні методи аналізу текстурних областей зображень засновані на тому, що текстура є фрактал - структуру, що складається з частин, які в якомусь сенсі подібні цілому. Таким чином, при визначенні фрактала використовується властивість самоподібності фрактала. Багато криві і поверхні статистично самоподобни, тобто кожна частина може вважатися зображенням цілого в зменшеному вигляді (рис. 3.1.3.1).

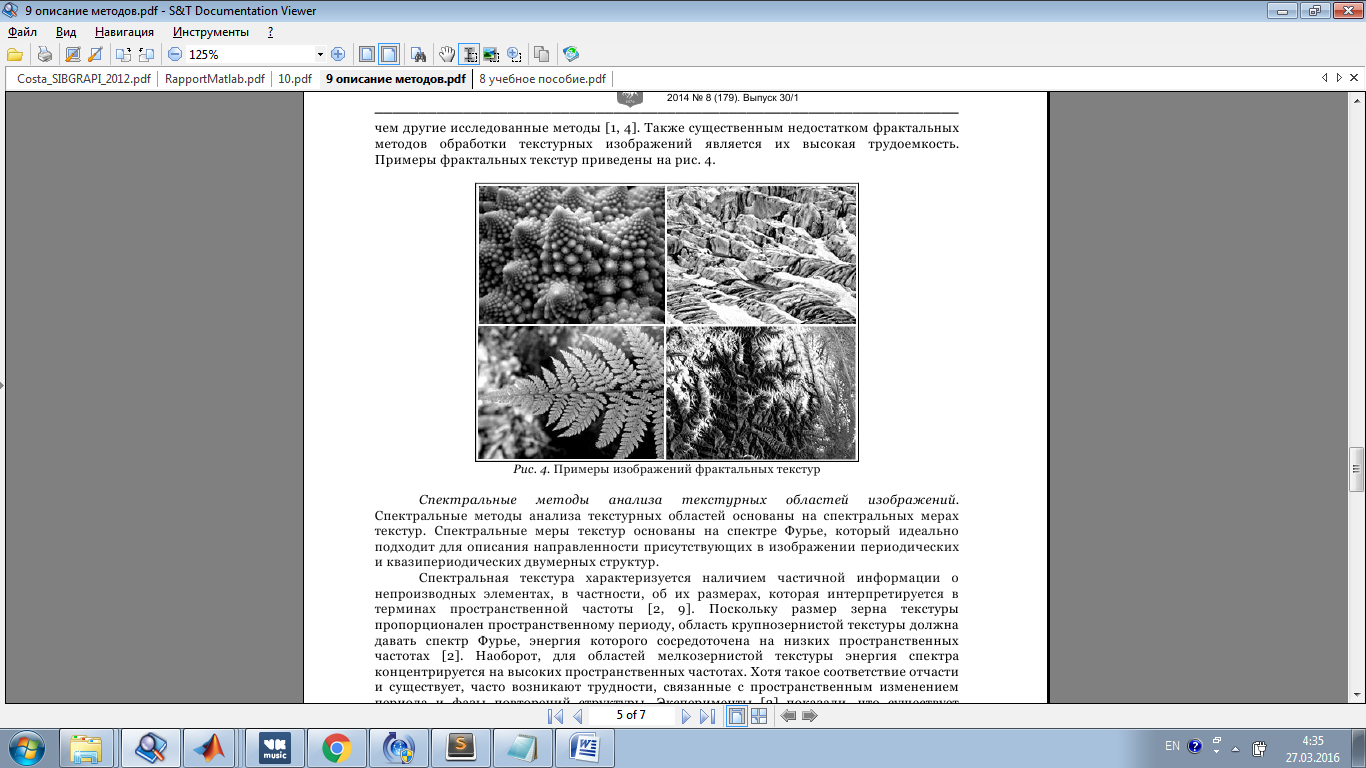


Рис. 3.1.3.1 Приклади зображень фрактальних текстур

Основною особливістю фрактальних характеристик є те, що фрактальні розмірності зображень штучних і природних об'єктів мають сильно розрізнятись. Це дозволяє успішно використовувати фрактальні ознаки для виявлення штучних змін ландшафту по фотографіях з космосу, виявлення штучних об'єктів на зображеннях, отриманих з телекамер і інших застосунків. Для сегментації використовуються характеристики, за якими обчислюється розмір фрактала.

Недоліком обробки фрактальних текстур є те, що, як правило, не можна побудувати глобальну фрактальную модель всього зображення. Загальна стратегія полягає в тому, щоб для знайдених опорних точок оцінити параметри локальних фрактальних моделей, зробити вирощування регіонів з близькими параметрами, класифікувати їх і провести сегментацію зображення. Також істотним недоліком фрактальних методів обробки текстурних зображень є їх висока трудомісткість.

## 3.1.4 Спектральні методи аналізу текстурних областей зображень.

Спектральні методи аналізу текстурних областей побудовані на спектральних мірах текстур. Спектральні міри текстур побудовані на спектрі Фур'є, який ідеально підходить для опису спрямованості присутніх в зображенні періодичних і квазіперіодичних двовимірних структур. Спектральна текстура характеризується наявністю часткової інформації про непохідні елементи, зокрема, про їх розміри, яка інтерпретується в термінах просторової частоти.

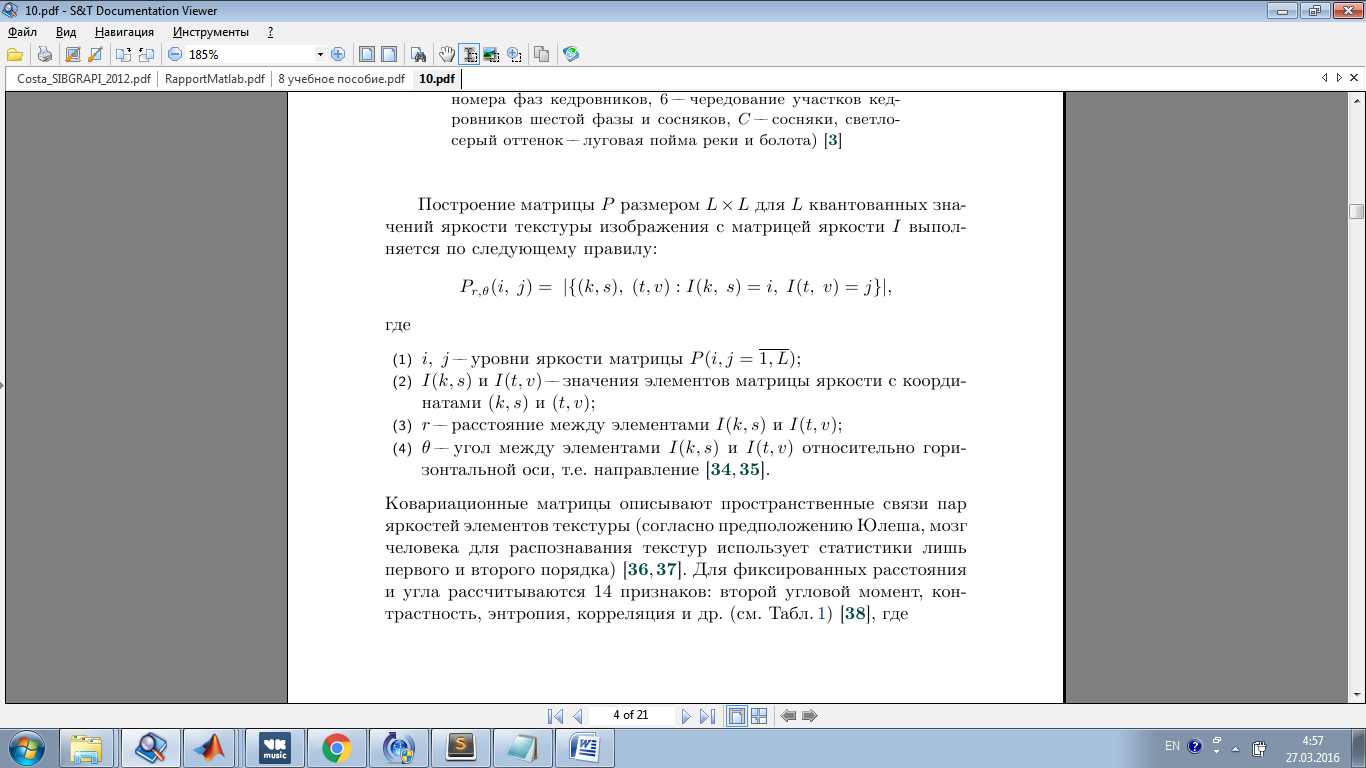
### 3.1.5 Комбіновані методи аналізу текстурних областей зображень.

Комбіновані методи засновані на використанні поєднання ознак різних груп. Наприклад, існує метод виділення областей на зображеннях з використанням фрактальних і текстурних характеристик знімків високого дозволу.

Далі ми більш деатльно розглянемо статистичний та фрактальний підходи.

## 3.2 Статистичний підхід до сегментації текстур

Виконується побудова матриці P розміром L\*L для квантованих значень яскравості текстури зображення з матрицею яскравості I за наступним правилом:

 (3.2.1)

де:

i, j – рівні яскравості атриці P;

I(k,s), I(t,v) – значення елементів матриці яскравості з відповідними координатами;

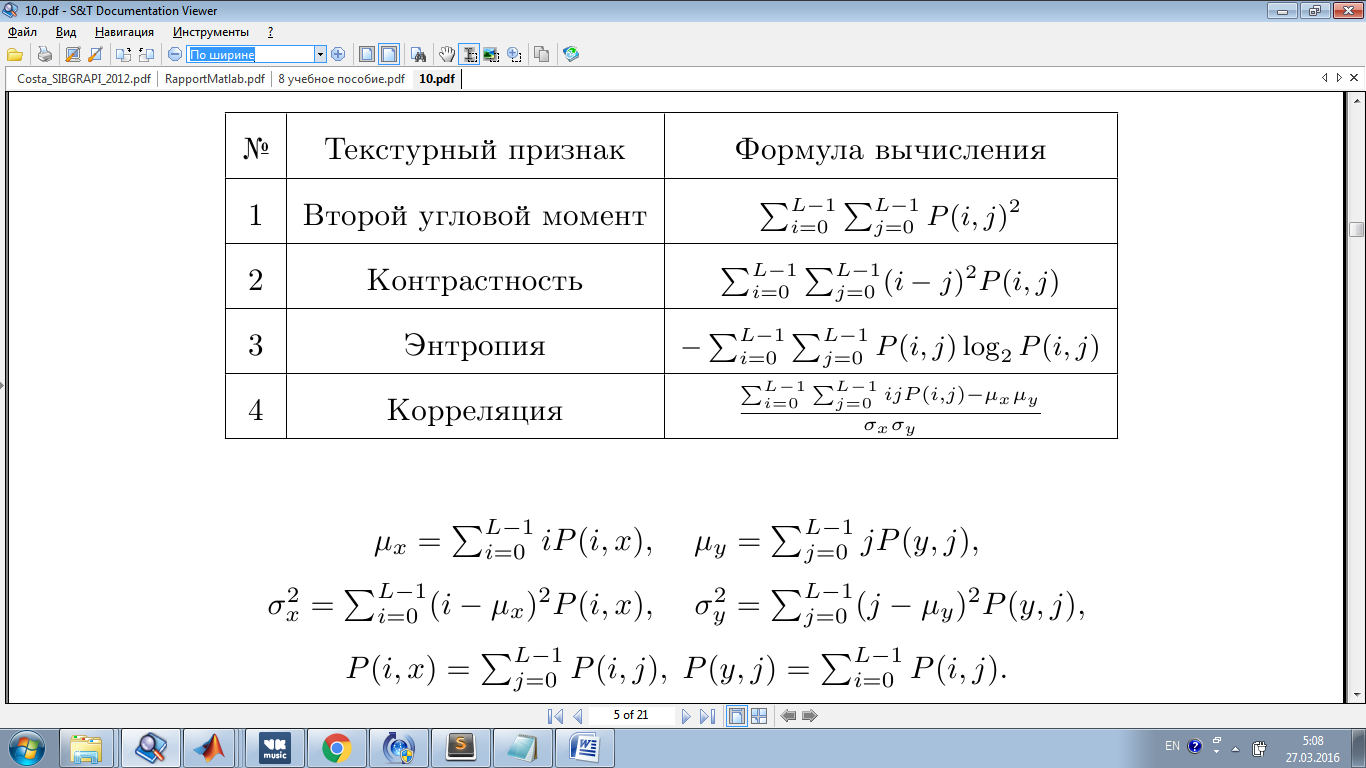
r – відстань між елементами I(k,s) та I(t,v);

θ – кут між елементами I(k,s) та I(t,v) відносно горизонтальної вісі.

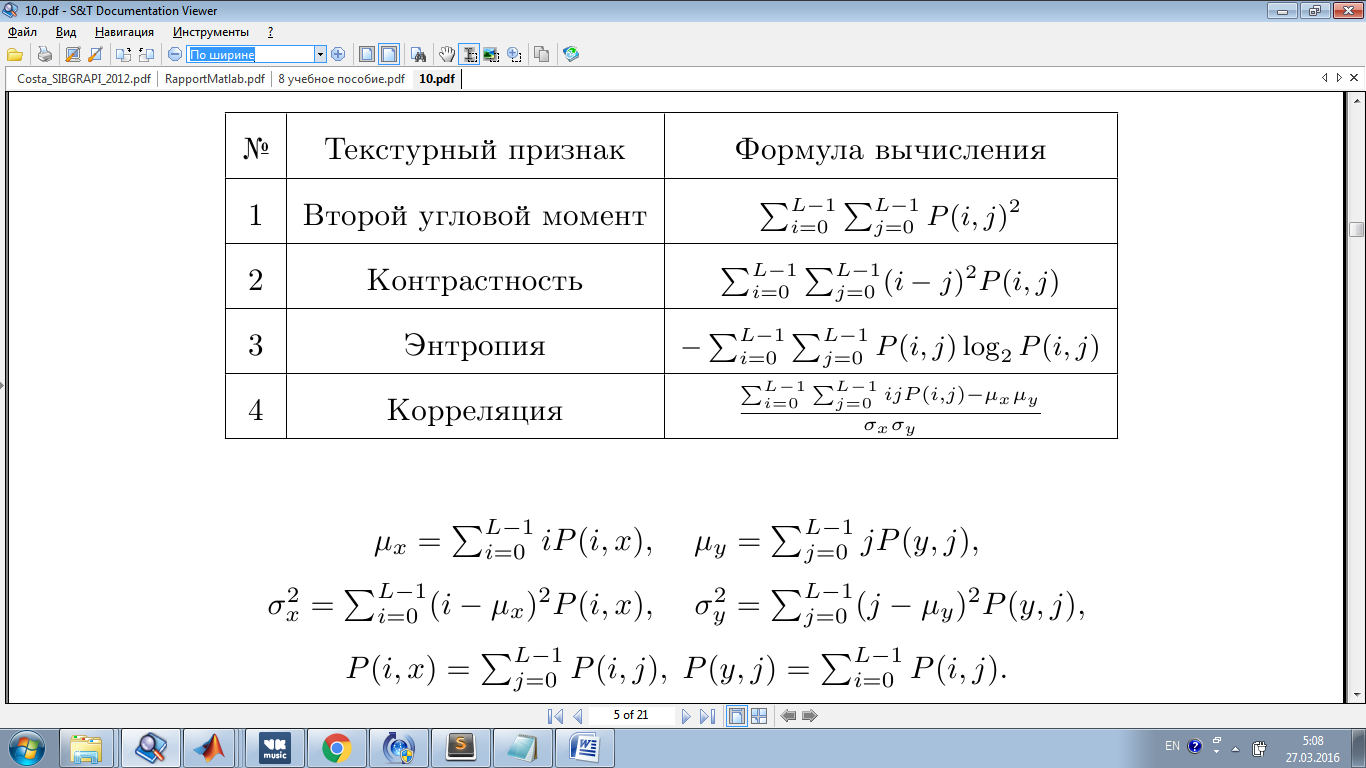
Коваріаційні матриці описують просторові зв'язки пар яркостей елементів текстури. Для фіксованих відстані і кута розраховуються 14 ознак: другий кутовий момент, контрастність, ентропія, кореляція і ін.

В методах, що реалізовані в пакеті MATLAB використовуються приведені нижче ознаки.

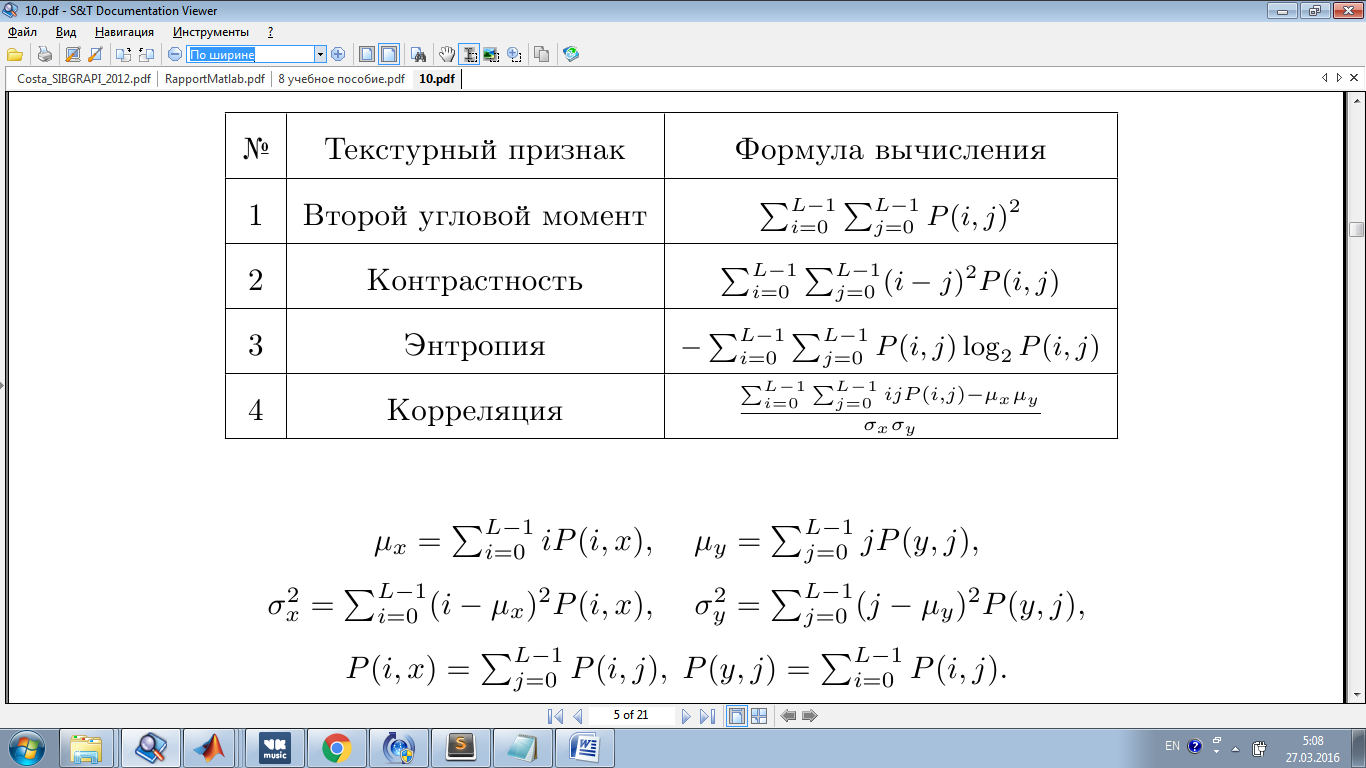
Другий кутовий момент:

 (3.2.2)

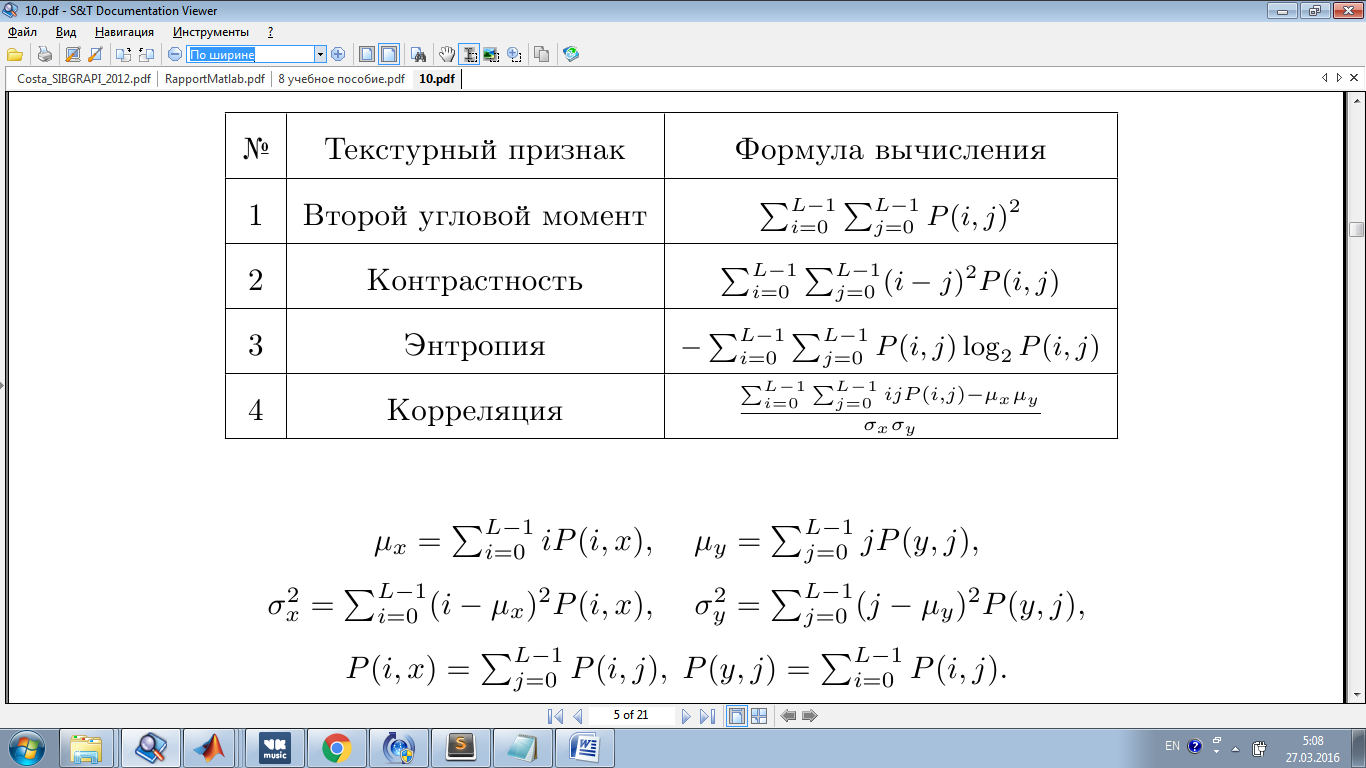
Контрастність:

 (3.2.3)

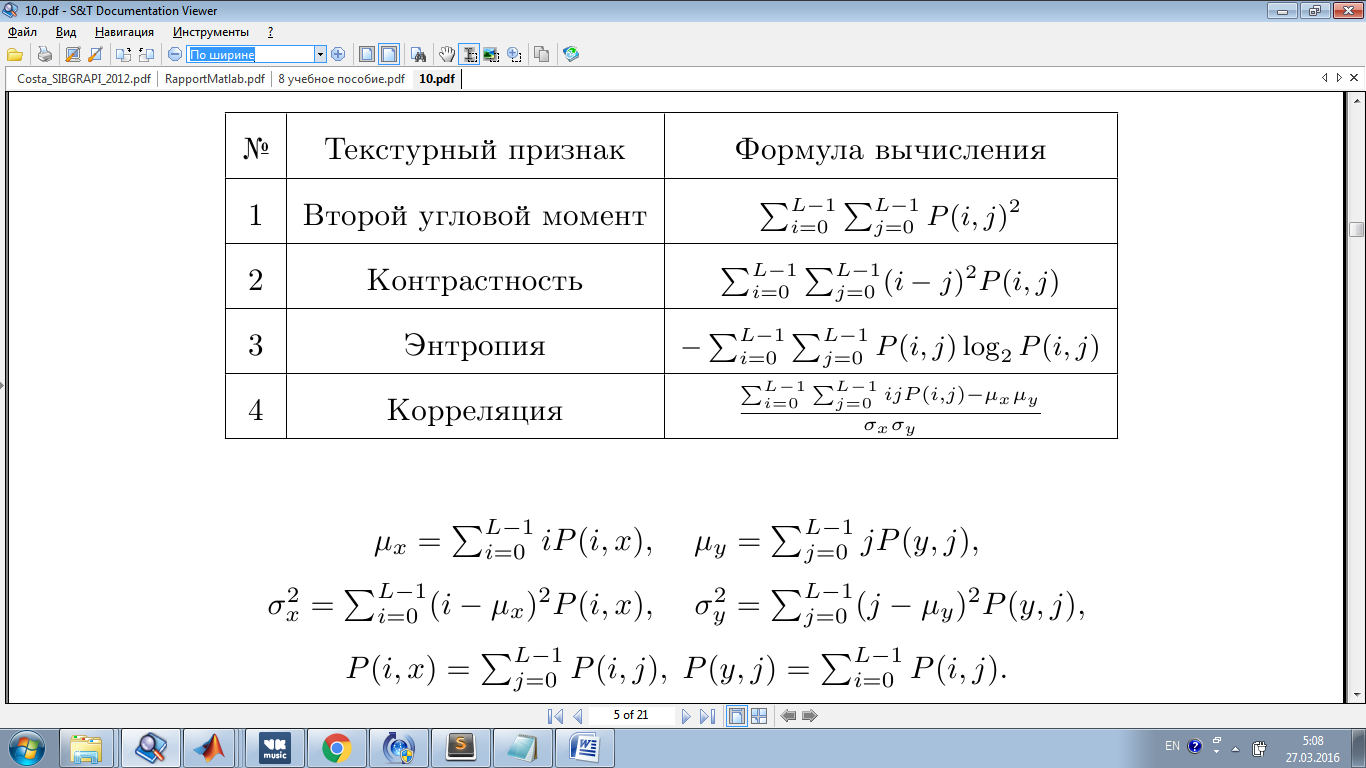
Ентропія:

 (3.2.4)

Кореляція:

 (3.2.5)

Для усіх циф формул необхідно прийняти:



## 3.3 Фрактальний підхід до сегментації текстур

### 3.3.1 Розмірність Хаусдорфа

*Розмірність Хаусдорфа* узагальнює поняття розмірності дійсного векторного простору, і є природним способом визначення розмірності підмножини в метричному просторі. Наприклад розмірність Хаусдорфа n-мірного (розмірність в сенсі векторного простору) унітарного простору (особливий випадок векторного простору) буде теж дорівнює n. Уявімо повне покриття множини X кулями радіуса не більше ніж г позначимо кількість цих куль за N (r) (рис. 5.3.1). Значення N (r) буде зростати при зменшенні r (для повного покриття потрібно буде дедалі більше куль). Розмірністю Хаусдорфа гладкої множини без особливостей (наприклад, фрактальних) X буде таке унікальне число d, що N (r) буде рости як 1/rd при наближенні r до нуля. Прикладами такихх множин можуть бути будь-які ідеалізовані геометричні об'єкти такі як куб, сфера і так далі.

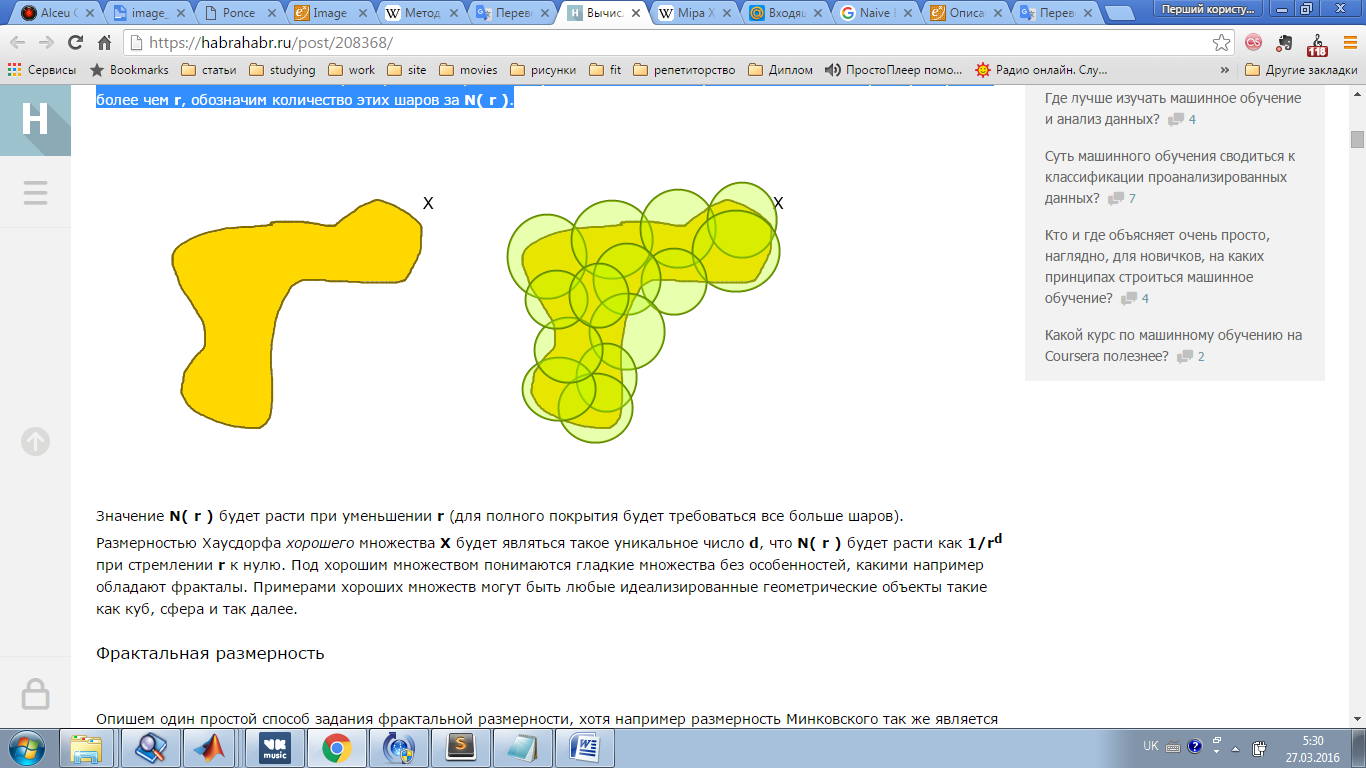


Рис. 3.3.1.1 Покриття множини кулями

[Ф. Хаусдорф](https://uk.wikipedia.org/wiki/%D0%A5%D0%B0%D1%83%D1%81%D0%B4%D0%BE%D1%80%D1%84_%D0%A4%D0%B5%D0%BB%D1%96%D0%BA%D1%81) розглядав деякий клас \mathcal{U} [відкритих множин](https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%B4%D0%BA%D1%80%D0%B8%D1%82%D0%B0_%D0%BC%D0%BD%D0%BE%D0%B6%D0%B8%D0%BD%D0%B0) X, на якому визначив невід'ємну функцію

l=\{l(A)\mid A\in\mathcal{U}\}  (3.3.1.1)

\lambda(B,\;\varepsilon)=\inf\left\{\sum_{i=1}^n l(A_i)\right\}, (3.3.1.2)

де нижня межа береться по всіх скінченних або зліченних [покриттях](https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%B4%D0%BA%D1%80%D0%B8%D1%82%D0%B5_%D0%BF%D0%BE%D0%BA%D1%80%D0%B8%D1%82%D1%82%D1%8F) [борелівської множини](https://uk.wikipedia.org/wiki/%D0%91%D0%BE%D1%80%D0%B5%D0%BB%D1%96%D0%B2%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BD%D0%BE%D0%B6%D0%B8%D0%BD%D0%B0) B\subset X множинами з \mathcal{U} з [діаметром](https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%BC%D0%B5%D1%82%D1%80), що не перевищує \varepsilon, тобто

B\subset\bigcup_{i=1}^n A_i\in\mathcal{U} (3.3.1.3)

\mathrm{diam}\,A_i\leqslant\varepsilon,\quad n=1,\;2,\;\ldots

Мірою Хаусдорфа \lambda, що визначається класом \mathcal{U} і функцією l, називається межа:

\lambda(B)=\lim_{\varepsilon\to 0}\lambda(B,\;\varepsilon). (3.3.1.4)

### 3.3.2 Фрактальна розмірність

Мандельброт надає таке визначення "фракталів називається структура, що складається з частин, які в якомусь сенсі подібні цілому". При визначенні фрактала використовується властивість його самоподібності. Багато кривих і поверхонь статистично самоподобні, тобто кожна частина може вважатися зображенням цілого в зменшеному вигляді.

Розмірність фрак тала D визначається як:

 (3.3.2.1)

де 1/r є відношення подібності, N - число кроків, необхідне для того, щоб покрити криву. На рис. 5.3.2 проілюстровано співвідношення між числом кроків і ставленням подібності на прикладі чотирьох ламаних прямих.

На практиці розмір фрактала для кривої оцінюється шляхом вимірювання довжин кривої при різних розмірах кроку. Розмірність фрактала D може бути оцінена за допомогою наступного рівняння регресії:

log L = C + B log G (3.3.2.2)

D = 1-В, (3.3.2.2)

де L - довжина кривої, B - нахил регресії, G - величина кроку, С - константа.

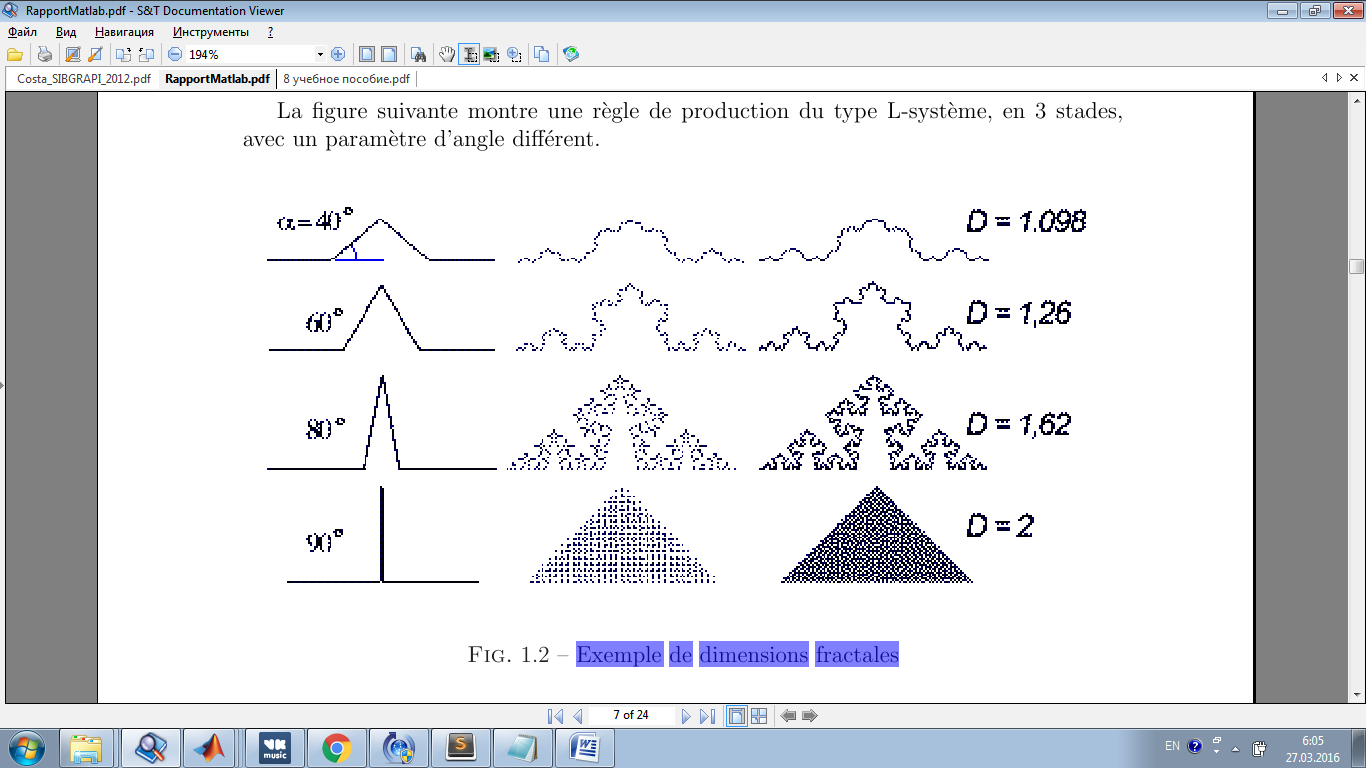


Рис. 3.3.2.1 Приклади фрактальних розмірностей

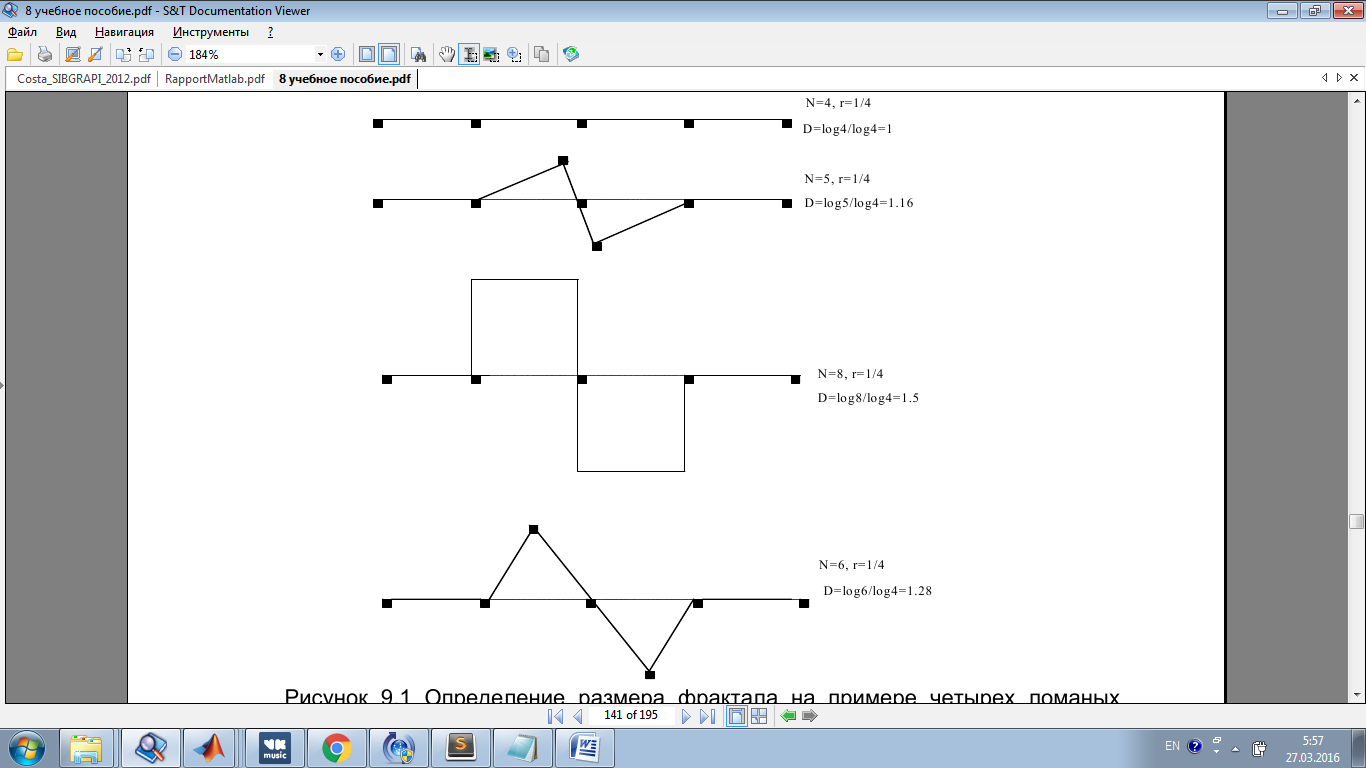
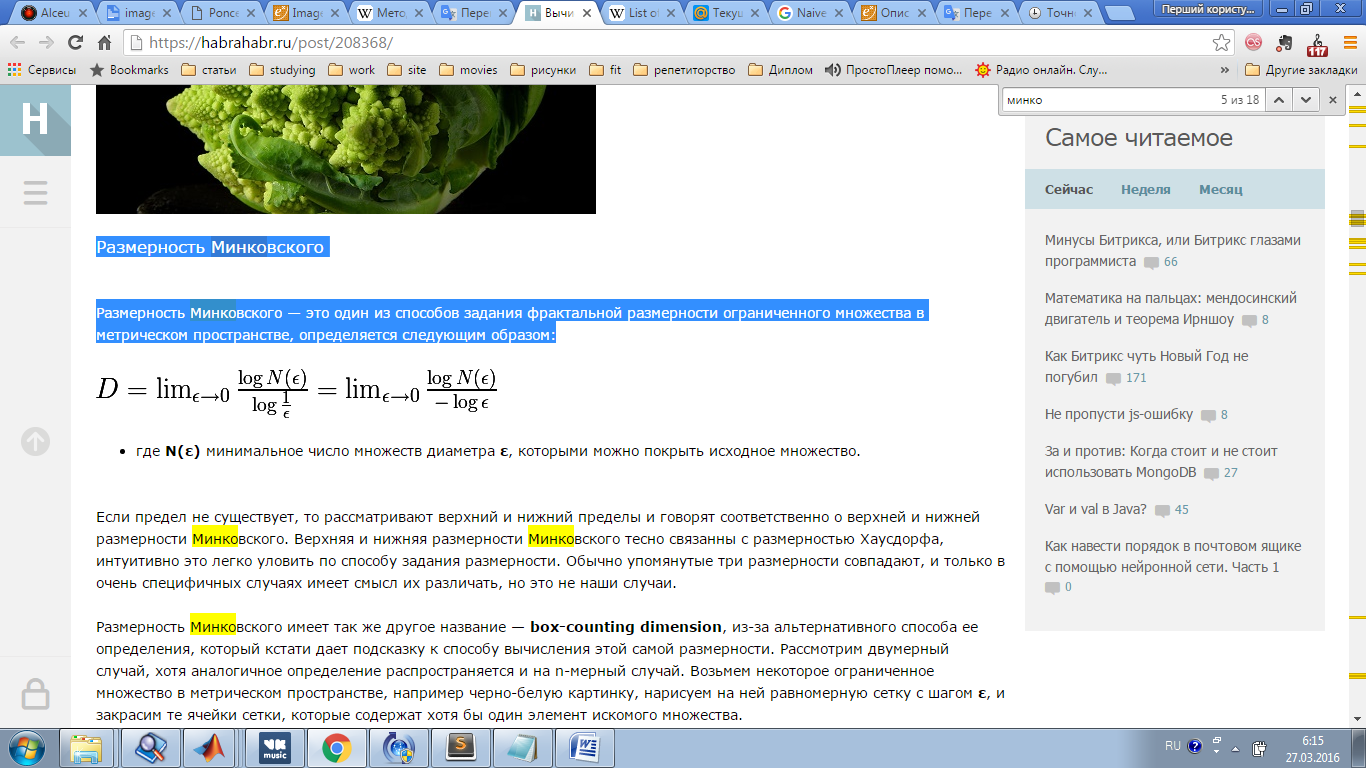
**

Рис. 3.3.2.3 Приклад визначення розмірності фракталу

### 3.3.3 Розмірність Мінковського

*Розмірність Мінковського* - це один із способів задання фрактальної розмірності обмеженого безлічі в метричному просторі, визначається наступним чином:

 (3.3.3.1)

N(ε) - мінімальне число множин діаметра ε, якими можна покрити вихідну множину.

Якщо межі не існує, то розглядають верхню і нижню межі і говорять відповідно про верхню та нижню розмірності Мінковського. Верхня і нижня розмірності Мінковського тісно пов'язані з розмірністю Хаусдорфа, інтуїтивно це легко зрозуміти за способом задання розмірності. Зазвичай згадані три розмірності збігаються, і тільки в дуже специфічних випадках має сенс їх розрізняти.

Розмірність Мінковського має так само іншу назвау - *box-counting dimension*, завдяки альтернативному способу її визначення. Розглянемо двовимірний випадок, хоча аналогічне визначення поширюється і на n-мірний випадок. Візьмемо деяку обмежену множину в метричному просторі, наприклад чорно-білу картинку, намалюємо на ній рівномірну сітку з кроком ε, і закрасимо ті осередки сітки, які містять хоча б один елемент шуканої множини. Далі почнемо зменшувати розмір осередків, тобто ε, тоді розмірність Мінковського буде обчислюватися за вищенаведеною формулою, досліджуючи швидкість зміни ставлення логарифмів.

### 3.3.4 Box-counting алгоритм.

Алгоритм виводиться в такий спосіб, позначимо за Dbc наближене значення розмірності Мінковського. Запишемо визначення цієї розмірності, прибравши межу, її ми будемо імітувати в ітераціях, в яких буде змінюватися розмір елементів.

 (3.3.4.1)

Якщо зафіксувати розміри осередків ε і розглядати Dbc як невідоме, то легко помітити, що наведене вираз є формулою лінії. Ми можемо запустити цикл з різних розмірах осередків ε і записувати результат. Якщо побудувати лінію регресії для отриманого безлічі даних, це значення і буде аппроксимацией фрактальної розмірності Мінковського.

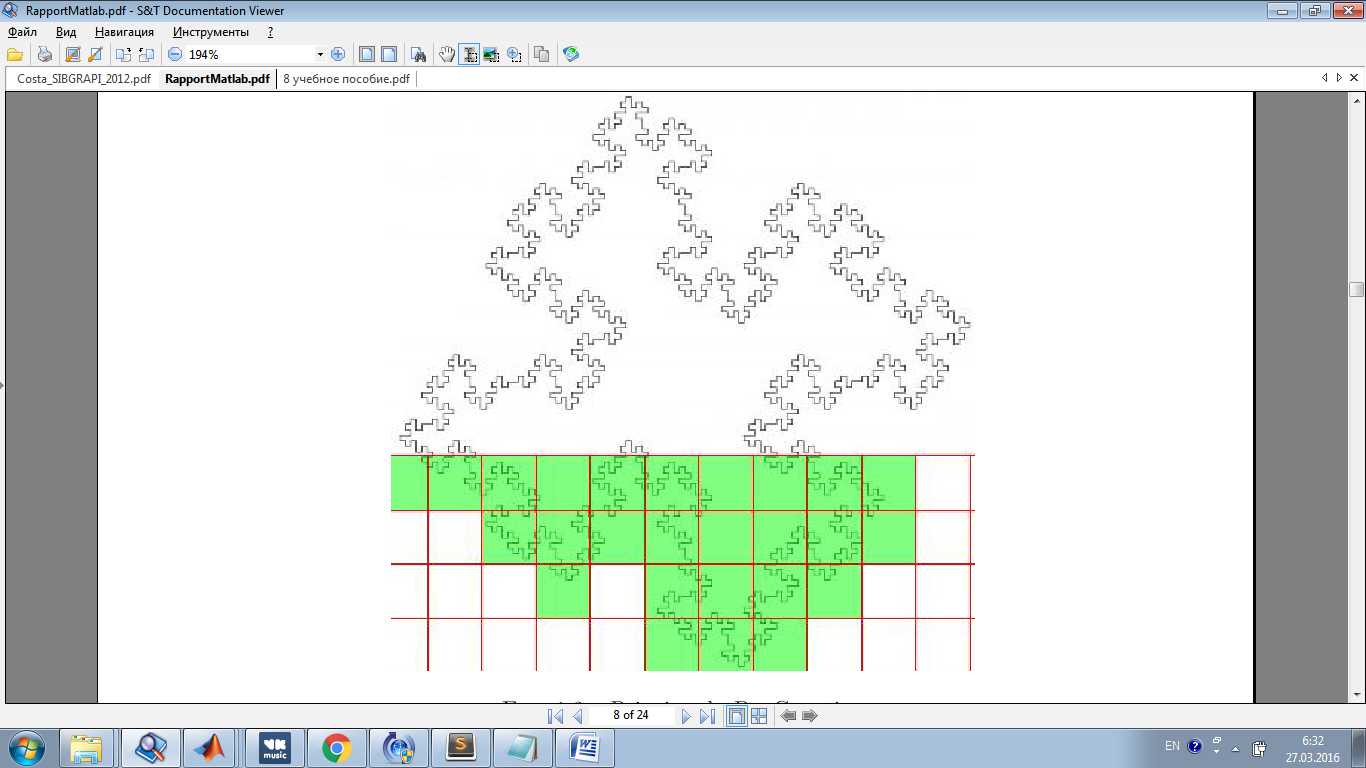


Рис. 3.3.4 Принцип box-counting алгоритму

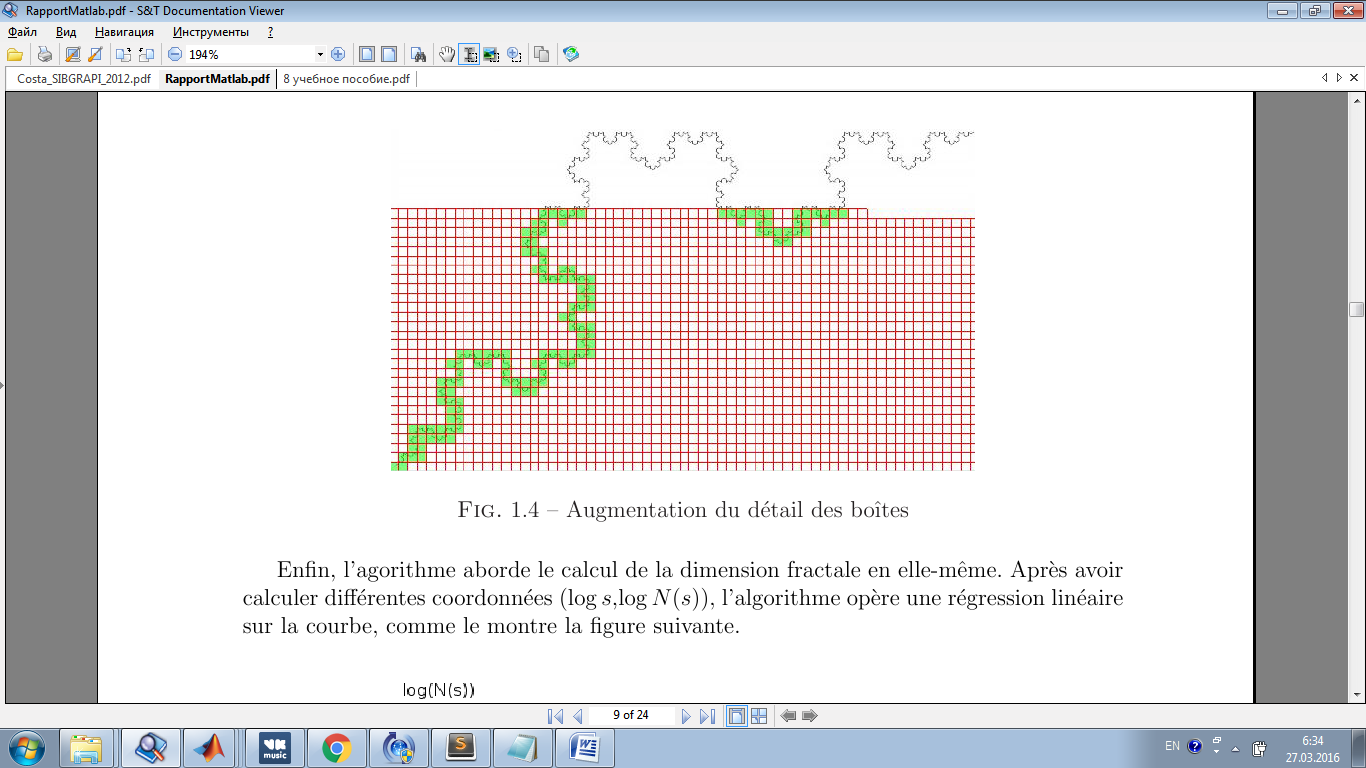


Рис. 3.3.4.1 Зменьшення розмірів елементів box-counting алгоритму

## 3.4 Метод Оцу обчислення порога бінарізації

*Метод Оцу* – це алгоритм обчислення порога бінарізації для напівтонового зображення, що використовується в області комп'ютерного розпізнавання образів і обробки зображень. Алгоритм дозволяє розділити пікселі двох класів ( "корисні" і "фонові") розраховуючи такий поріг, щоб внутриклассовая дисперсія була мінімальною. Метод Оцу також має поліпшену версію для підтримки декількох рівнів зображення, який отримав назву мульти-Оцу метод.

Метод Оцу шукає поріг, що зменшує дисперсію всередині класу, яка визначається як зважена сума дисперсій двох класів:

\sigma^2_w(t)=\omega_1(t)\sigma^2_1(t)+\omega_2(t)\sigma^2_2(t), (3.4.1)

де ваги \omega_i — це ймовірності двох класів, розділених порогом t,  
\sigma^2_i — дисперсія цих класів.

Оцу показав, що мінімізація дисперсії всередині класу рівноцінна максимізації дисперсій між класами:

\sigma^2_b(t)=\sigma^2-\sigma^2_w(t)=\omega_1(t)\omega_2(t)\left[\mu_1(t)-\mu_2(t)\right]^2 (3.4.2)

яка виражається в термінах ймовірності \omega_i і середнього арифметичного \mu_i класу, яке, в свою чергу, може оновлюватися ітеративно. Ця ідея привела до ефективного алгоритму.

Нижче наведемо сам алгоритм.

Нехай дано монохромне зображення G(i,j), i=\overline {1, Height}, j=\overline {1, Width}. Лічильник повторювань k=0.

1. Обчислити гістограму p(l) зображення і [частоту](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BD%D0%BE%D1%81%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%87%D0%B0%D1%81%D1%82%D0%BE%D1%82%D0%B0_%D1%80%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B9_%D1%8D%D0%BA%D1%81%D0%BF%D0%B5%D1%80%D0%B8%D0%BC%D0%B5%D0%BD%D1%82%D0%B0) N(l) для кожного рівня інтенсивності зображення G.

2. Обчислити початкові значення для \omega_1(0), \omega_2(0) и \mu_1(0), \mu_2(0).

3. Для кожного значення t = \overline {1, max(G)} – полутона – горизонтальна вісь гістограми:

3.1. Оновлюємо \omega_1, \omega_2 та \mu_1, \mu_2

3.2. Обчислюємо

\sigma^2_b(t)=\omega_1(t)\omega_2(t)\left[\mu_1(t)-\mu_2(t)\right]^2. (3.4.3)

3.3. Якщо \sigma^2_b(t) більше, ніж наявне, то запам'ятовуємо \sigma^2_b і значення порогу t.

4. Шуканий поріг відповідає максимуму \sigma^2_b(t).

 N_T = \sum^{max(G)}_{i=0} {p(i)} = Height \cdot Width, (3.4.4)

\omega_1(t) = {{\sum^{t-1}_{i=0} {p(i)}} \over {N_T}} = {\sum^{t-1}_{i=0} {N(i)}},\quad \omega_2(t) = 1 - \omega_1(t), (3.4.5)

\mu_T = {{\sum^{max(G)}_{i=0} {i \cdot p(i)}} \over {N_T}} = {\sum^{max(G)}_{i=0} {i \cdot N(i)}} , (3.4.6)

\mu_1(t) = {{\sum^{t-1}_{i=0} {i \cdot p(i)}} \over {N_T \cdot \omega_1(t)}} = {{\sum^{t-1}_{i=0} {i \cdot N(i)}} \over {\omega_1(t)}}, \quad \mu_2(t) = {{\mu_T - \mu_1(t) \cdot \omega_1(t)} \over {\omega_2(t)}}.(3.4.7)

## 3.5 Наївний Баєсів класифікатор

Наївний баєсів класифікатор - простий імовірнісний класифікатор, заснований на застосуванні теореми Баєса зі строгими (наївними) припущеннями про незалежність.

Залежно від точної природи ймовірнісної моделі, наївні баєсові класифікатори можуть навчатися дуже ефективно. У багатьох практичних додатках для оцінки параметрів для наївних Баєсових моделей використовують метод максимальної правдоподібності.

Перевагою наївного байєсівського класифікатора є мала кількість даних для навчання, необхідних для оцінки параметрів, необхідних для класифікації.

Імовірнісна модель для класифікатора - це умовна модельp(C \vert F_1,\dots,F_n)\, над залежною змінною класу C з малою кількістю результатів або класів, залежна від кількох змінних F_1 … F_n. Проблема полягає в тому, що коли кількість властивостей дуже велике або коли властивість може приймати велику кількість значень, тоді будувати таку модель на імовірнісних таблицях стає неможливо. Тому ми переформулюємо модель, щоб зробити її зручною для обробки.

Викристовуючи [теорему Байеса](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%91%D0%B0%D0%B9%D0%B5%D1%81%D0%B0), запишемо

p(C \vert F_1,\dots,F_n) = \frac{p(C) \ p(F_1,\dots,F_n\vert C)}{p(F_1,\dots,F_n)}. \, (3.5.1)

На практиці цікавий лише чисельник цього дробу, так як знаменник не залежить від C і значення властивостей F_i дані, так що знаменник - константа.

Чисельник еквівалентний спільної ймовірності моделі p(C, F_1, \dots, F_n)\,, яка може бути переписана наступним чином, використовуючи повторні додатки визначень умовної ймовірності:

p(C, F_1, \dots, F_n)\,

= p(C) \ p(F_1,\dots,F_n\vert C)

= p(C) \ p(F_1\vert C) \ p(F_2,\dots,F_n\vert C, F_1)

= p(C) \ p(F_1\vert C) \ p(F_2\vert C, F_1) \ p(F_3,\dots,F_n\vert C, F_1, F_2) (3.5.2)

= p(C) \ p(F_1\vert C) \ p(F_2\vert C, F_1) \ \dots p(F_n\vert C, F_1, F_2, F_3,\dots,F_{n-1})

Тепер можна використовувати "наївні" припущення умовної незалежності: припустимо, що кожна властивістьF_i умовно незалежно від будь-якої іншої властивості F_j при j\neq i. Це означає:

p(F_i \vert C, F_j) = p(F_i \vert C)\, (3.5.3)

Таким чином, спільна модель може бути виражена як:p(C, F_1, \dots, F_n)
= p(C) \ p(F_1\vert C) \ p(F_2\vert C) \ p(F_3\vert C) \ \cdots\, (3.5.4)

= p(C) \prod_{i=1}^n p(F_i \vert C).\,

Це означає, що з припущення про незалежність, умовний розподіл класової змінної може бути виражено так:

p(C \vert F_1,\dots,F_n) = \frac{1}{Z}  p(C) \prod_{i=1}^n p(F_i \vert C), (3.5.5)

де Z — це масштабний множник, що залежить тільки від F_1,\dots,F_n, тобто константа, якщо значення змінних відомі.

Наївний байесовский класифікатор об'єднує модель з правилом рішення. Одне загальне правило має вибрати найбільш ймовірну гіпотезу; воно відомо як апостеріорное правило прийняття рішення (MAP). Відповідний класифікатор - це функція \mathrm{classify}, визначена в такий спосіб:

\operatorname{classify}(f_1,\dots,f_n) = \arg \max_c p(C=c) \prod_{i=1}^n p(F_i=f_i\vert C=c) (3.5.6)

## 3.6 Висновки

Можна сформулювати наступні особливості алгоритмів сегментації текстурних зображень:

* фрактальні алгоритми краще використовувати для зображень зі складною структурою текстур;
* фрактальний алгоритм з підрахуванням розмірності Мінковського краще використовувати для зображень з високою якістю;
* статистичні алгоритми краще використовувати для зображень з простою структурою текстур, з яскраво вираженими межами між текстурами.

Для розроблюваної програми важлива працездатність алгоритм в режимі, коли чітка границя між текстурами візуально погано помітна (наприклад у сутінках, або коли узбіччя може за кольором мао відрізнятись від дороги), тому використання саме фрактального піходу до текстурного аналізу є критично важливим.

Доцільність використання розмірності Хаусдорфа або Мінковського залежить від якості зобраеня, що подається на обробку програми, тож необхідно реалізувати обидва методи, а поріг обрання першого чи другого буде визначено експерементально за тестовою вибіркою.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Coggins J.M. A Framework for Texture Analysis Based on Spatial Filtering Ph. D. // Computer Science Department. – Michgan: Michigan State University, 1982.
2. Haralick R.M. Statistical and Structural Approaches to Texture // Proc. of the IEEE, 67. – 1979. – P. 786-804.
3. Mark S. Nixon Feature Extraction and Image Processing / Mark S. Nixon, Alberto S. Aguado. – Oxford: A division of Reed Educational and Professional Publishing Ltd, 2002. – 330 p.
4. Schlesinger M.I., and Flach B., Some Solvable Subclasses of Structural Recognition Problems. Proc. of Czech P at- tern Recognition Workshop: 55–62 (2000).
5. Tamura H., Mori S., Yamawaki Y. Textural Features Corresponding to Visual Perseption // IEEE Transactions on Systems, Man and Cybernetics. – 1978. – № 8. – P. 460-473.
6. Tuceryan M., Jain A.K. Texture Segmentation Using Voronoi Polygons // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1990. – № 12. – P. 211-216.
7. Wainwright M.J., and Jordan M.I., Graphical Models, Exponential Families, and Variational Inference. Founda- tions and Trends in Machine Learning. 1: 1–305 (2008).
8. Гонсалес Р., Вудс Р., Эддинс С. Обработка изображений в среде Matlab. – М.: Техносфера, 2006. – 616 с.
9. Двоенко С.Д., Копылов А.В., Моттль В.В. Задача рас- познавания образов в массивах взаимосвязанных объ- ектов. Алгоритм распознавания// Автоматика и теле- механика. - 2005. № 12. - С. 162-176.
10. Дунаев А. А., Лобив И. В., Мехонцев Д. Ю., Мурзин Ф. А., Половинко О. Н., Семич Д. Ф., Чепель А. В., Ярков К. А. Алгоритмы быстрого поиска фрагмен- тов фотографических изображений // Современные проблемы конструирования программ. — Новосибирск, 2002. — С. 88 — 110.
11. Морозов А.Д. Введение в теорию фракталов / А.Д. Морозов. – М.: ―Современная математика‖, 2002. – 57 с.
12. Прэтт У. Цифровая обработка изображений: [в 2-х томах] / У. Прэтт. – М.: Мир, 1982. – Т. 1. – 312 с.
13. Прэтт У.К., Фожра О.Д., Гагалович А. Применение моделей стохастических текстур для обработки изображений // ТИИЭР. — 1981. — Т. 69, № 5.
14. Фисенко В.Т. Фрактальные методы сегментации текстурных изображений / В.Т. Фисенко, Т.Ю. Фисенко // Приборостроение, 2013. – Т. 56 № 5. – С. 63–70
15. Довідка математичного пакету MATLAB [Електронний ресурс]. – Режим доступу: <http://www.mathworks.com/>
16. Wikitude Drive – навигатор с дополненной реальностью Електронний ресурс]. – Режим доступу: http://www.doroga78.ru/advice/799/