

УДК 004.3

Н.С. Костюкова, С.О. Шеховцов  
Донецький національний технічний університет  
shozda@r5.dgtu.donetsk.ua**Контентно – залежне масштабування зображень зі збереженням значущих об'єктів**

*У даній роботі була проведена порівняльна оцінка алгоритмів масштабування за кількома найбільш важливими характеристиками, виконана класифікація алгоритмів за деякими ознаками, були виявлені їх основні недоліки. Для оптимізації масштабування був запропонований метод контентно-залежного масштабування, який змінює зображення в залежності від його вмісту. Показано, що цей метод має значну перевагу перед іншими за рахунок меншого викривлення зображення та можливості виділення областей, які взагалі не слід масштабувати.*

**Масштабування зображень, контентно- залежне масштабування****Вступ**

Сучасні електронні інформаційні ресурси значною мірою представлені у вигляді колекцій зображень та відео даних. При доступі до таких ресурсів існує декілька проблем, серед яких однією з найважливіших є масштабування, або зміна розмірів зображення. Задача масштабування особливо актуальна при перенесенні зображення на мобільні пристрої, де воно не може відображатися в повному розмірі, а також при розміщенні зображень на сайтах в мережі Internet. Збільшення зображень часто застосовується для більш детального огляду фрагмента зображення, при обробці зображень. В роботі розглядається модифікований алгоритм контентно- залежного масштабування зображень, виконується його оцінка з точки зору ступеня спотворення зображення і часу виконання

**Задача контентно- залежного масштабування зображення**

В загальному вигляді задача контентно-залежного масштабування зображення може бути сформульована наступним чином.

Нехай  $X$  – вхідне зображення, що є матрицею  $A \times B$ , кожен елемент якої описує один піксель зображення трьома компонентами кольору.  $A$  – кількість стовпців матриці,  $B$  – кількість рядків. Тоді  $X'$  – результат застосування до вхідного зображення функції масштабування  $f$ ,  $X' = f(X)$ .

Функція  $f$  виконує видалення (дублювання) незначних ділянок зображення для надання йому нових, зазначених користувачем, розмірів без спотворення вмісту зображення. З

цього походить, що функція масштабування зводиться до знаходження незначних ділянок зображення і подальшого використання виявлених ділянок для трансформації.

Ділянкою для модифікації є безперервний ланцюжок товщиною в один піксель з мінімальною сумарною енергією, тому для масштабування зображення необхідно обчислення енергій всіх його точок.

Обчислення енергії відбувається за наступною формулою:

$$E = e(x_{ij}),$$

де  $x_{ij}$  – піксель зображення, розташований в  $i$ -ому стовпці та  $j$ -ому рядку. Енергія  $e(x_{ij})$  представляє собою функцію, залежну від схожості крапки на сусідні точки і власне відображає цю схожість.

**Аналіз досліджень у галузі масштабування зображень**

При масштабуванні використовуються різні алгоритми. Для векторних зображень масштабування відбувається без втрат якості зображення, а для растрових має місце втрата якості зображення, особливо при зміні розміру без збереження пропорцій [1].

Пошук ефективних способів реалізації масштабування растрових зображень без видимих втрат якості є предметом досліджень багатьох учених різних країн світу. До них належать Джефф Етвуд (колишній співробітник Vertigo Software, Поїнт Річмонд, Каліфорнія) [2], Шай Авідан (співробітник Mitsubishi Electric Research Labs, Північна Америка) [3], Аріель Шамір (Техаський університет в Остині) [4]. Всі алгоритми, залежно від способу дії, умовно можна

розділити на чотири категорії: алгоритми з копіюванням, інтерполюючі алгоритми, піксельні та алгоритми на основі предискретизації. До алгоритмів, що використовують копіювання, відноситься інтерполяція методом найближчого сусіда. До алгоритмів, що інтерполюють, відносяться білінійна і бікубічна інтерполяція. Категорія піксельних алгоритмів включає алгоритми EPX/Scale2x/AdvMAME2x, Scale3x/AdvMAME3x, Eagle, 2xSaI, Super 2xSaI і Super Eagle, сімейство hqnx [5]. Четверта категорія – на основі предискретизації – залежно від методу обробки зображення ділиться ще на три підкатегорії: з цілим коефіцієнтом (децимації, інтерполяція і зміна частоти дискретизації), за допомогою поліфазних фільтрів (фільтри інтерполяційного типу, фільтри з дзвоноподібною характеристикою, віконні sinc-фільтри) і за допомогою дискретного перетворення Фур'є. Класифікація алгоритмів масштабування наведена на рис.1.

Порівняння алгоритмів масштабування, виконане за такими характеристиками, як витрати пам'яті, часові витрати і спотворення зображенні (оцінює, наскільки погіршується якість зображення при його збільшенні або зменшенні), показало, що всі розглянуті алгоритми є малоефективними.

Так, наприклад, метод найближчого сусіда, який витрачає найменше часу і пам'яті на виконання, дає дуже велике спотворення зображень при зміні їхнього розміру. Алгоритми предискретизації, які дають хороші показники якості зображення, вимагають значних часових і просторових ресурсів. До того ж при

масштабуванні на зображеннях спостерігається ефект розмиття. Білінійна інтерполяція, хоч і вимагає відносно невеликих витрат пам'яті / часу, сильно спотворює зображення на виході. Бікубічна інтерполяція, як удосконалення білінійної, витрачає приблизно стільки ж ресурсів, спотворює зображення менше, але вже призводить до менших спотворень.



Рисунок 1 – Класифікація алгоритмів масштабування зображень

Також оцінювалось викривлення зображення, що характеризує видиму втрату якості при масштабуванні. Викривлення залежить від вибору алгоритму, який обчислює енергію; порівняльний аналіз для деяких з цих алгоритмів можна виконати на підставі рис.2.

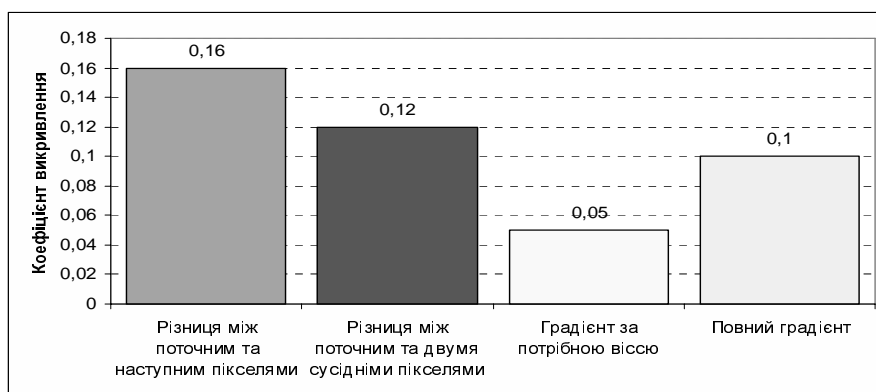


Рисунок 2 – Залежність ступеню викривлення зображення від способу знаходження енергії пікселя

З рисунку 2 видно, що найменший коефіцієнт викривлення притаманний алгоритму, що застосовує градієнт за потрібною віссю. Найменш бажано вживати обчислення на основі різниці між поточним та наступним пікселем, оскільки на виході можна отримати велике викривлення зображення.

Отже, розробка алгоритмів контентного масштабування зображень, які б дозволили зменшити спотворення зображень без збільшення витрат часу і пам'яті, є актуальною задачею. Мета статті полягає в описі запропонованого авторами модифікованого алгоритму контентного масштабування і його оцінці з точки зору

спотворення зображень і часових характеристик реалізації.

### **Модифікований алгоритм контентного масштабування зображень**

Проведений аналіз алгоритмів масштабування показав їх високу ступінь спотворення зображень. Ще однією характерною рисою переважної більшості алгоритмів є те, що в них не враховуються особливості змісту зображення. Проте існує клас задач, в яких бажано зберегти головний об'єкт зображення без істотної деформації.

Одним з варіантів реалізації масштабування зі збереженням головного об'єкту є знаходження контуру одним з відомих алгоритмів виділення контурів, що призведе до виключення знайденого контуру з процесу масштабування і виконання масштабування на зображенні, що залишилось. Проте наведений метод має два принципових недоліки. Перш за все, обчислювальна складність виділення контуру занадто велика та не дозволяє на існуючих апаратних засобах реалізувати масштабування за час, наближений до реального. По-друге, в результаті отримується зображення, спотворене через велику розбіжність між пропорціями масштабованого фону та збереженого в оригінальному розмірі об'єкту.

Ці недоліки відсутні в алгоритмі Seam Carving, розробленому в 2007 році співробітниками Mitsubishi Electric Research Labs (MERL) Шаї Авіданом і Аріелем Шаміром [10]. Саме тому базовим для модифікацій обрано алгоритм Seam Carving, який може використовуватись для вертикального або горизонтального контентно-залежного масштабування. В алгоритмі використовується поняття ланцюжка пікселів. Ланцюжок – це такий набір пікселів, що в кожному рядку вибрано рівно по одному пікселю, і сусідні пікселі в ньому поєднані сторонами або кутами [7]. Алгоритм полягає у виконанні наступних кроків.

Крок 1. Знаходження енергії кожної точки. На цьому етапі необхідно визначити, які частини зображення є більш важливими, а які – менш важливими, виходячи з отриманих даних, далі буде змінюватись розмір зображення [6].

Крок 2. Знаходження такої вертикального ланцюжка пікселів, щоб сумарна енергія пікселів, які входять в цей ланцюжок була мінімальною. При знаходженні такого ланцюжка можливе видалення його із зображення, мінімально зачіпаючи інформаційне наповнення зображення.

Крок 3. Після знаходження ланцюжка з мінімальною енергією, необхідно його видалити, якщо виконується зменшення зображення, або розтягнути, якщо виконується збільшення [6].

Модифікований алгоритм враховує вміст зображення: спочатку оцінюється важливість різних ділянок зображення, а потім змінюється розмір лише тих частин зображення, що були оцінені як неважливі.

### **Спотворення зображень при застосуванні модифікованого алгоритму контентно-залежного масштабування**

Було проаналізовано роботу алгоритму на різних класах зображень. Перший клас – це зображення, на яких чітко можна розділити об'єкт та фон (див.рис.3, а)). Результати виконання алгоритму для зменшення розміру зображення показані на рис. 3, б), в).

Як видно з цього прикладу, алгоритм обрав в якості незначних ділянок елементи фону зображення, а сам об'єкт залишився без значних змін, наскільки це дозволяв ступінь масштабування. Ланцюжки пікселів, які алгоритмом вважаються незначними, показані на рис. 4.

Другий клас тестових зображень – це зображення, на яких об'єкти мають однорідний колір. В цьому випадку алгоритм може помилятися, вважаючи частини об'єктів такими, що належать до фону, і видаляючи їх. Це може приводити до викривлення зображення (приклад див. на рис. 5).

Третій клас зображень, що використовувались для тестування алгоритму – це зображення, на яких складно виділити фон, які містять багато об'єктів, що перетинаються і торкаються одне одного. В таких випадках алгоритм буде давати значні викривлення за рахунок змінення самих зображень (приклад див. на рис. 6). Зокрема, при зменшенні по горизонталі суттєво змінилася форма дверей та інших вертикальних елементів. У наведеному прикладі викривлення відбувається тільки в разі горизонтального зменшення, бо при вертикальному алгоритм видаляє спочатку ланцюжки, що відповідають стіні на зображенні (тобто власне елементи фону), а зображення людей (об'єкти) залишає з якомога меншими змінами. Ланцюжки, що підлягають трансформації, відображені на рис. 7.

Всі наведені вище приклади відповідають випадку зменшення зображення. При збільшенні зображення так само спостерігається викривлення, оскільки для масштабування в обидві сторони (зменшення/збільшення) використовуються одні й ті самі ланцюжки мінімальної енергії, тільки для збільшення вони не видаляються, як для зменшення, а дублюються.

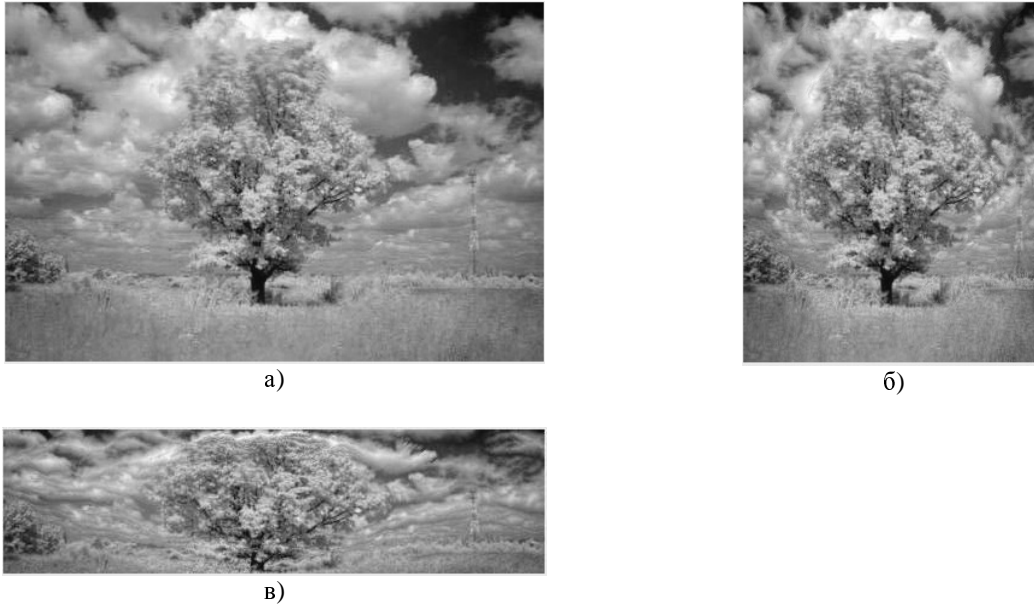


Рисунок 3 – Результати контентного масштабування зображень класу 1:  
а) вихідне зображення, б) результат зменшення по горизонталі, в) результат зменшення по вертикалі



Рисунок 4 - Ланцюжки, обрані для масштабування: а) по горизонталі; б) по вертикалі

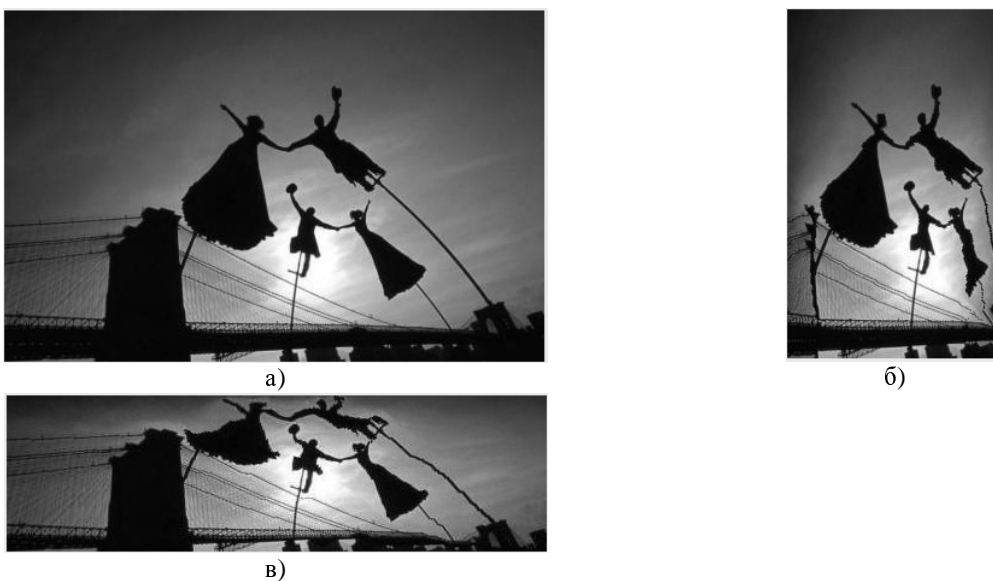


Рисунок 5 – Результати контентного масштабування зображень класу 2:  
а) вихідне зображення, б) результат зменшення по горизонталі, в) результат зменшення по вертикалі



а)



б)



в)

Рисунок 6 – Результати контентного масштабування зображень класу 3:  
а) вихідне зображення, б) результат зменшення по горизонталі, в) результат зменшення по вертикалі



а)



б)

Рисунок 7 - Ланцюжки, обрані для масштабування: а) по горизонталі; б) по вертикалі

Для оцінки результатів тестування алгоритму на різних класах зображень оцінимо коефіцієнт збігу силуетів оригінального зображення та зменшеного на 30% (див.рис.8). До класу 1 віднесені зображення з однорідним фоном об'єктів; до класу 2 – зображення, де чітко можна розділити фон та об'єкт; до класу 3 – зображення, де об'єкти накладаються один на один)

Слід зазначити, що ланцюжки з мінімальною енергією обчислюються не один раз,

а після кожного змінення розміру зображення, оскільки при масштабуванні ці ланцюжки змінюються.

Після проведення експериментів на різних зображеннях, було виявлено загальну тенденцію, що при наявності ярко вираженого об'єкту, що знаходиться в центрі зображення, алгоритм намагається обирати для модифікації ланцюжки, що огинають цей об'єкт, як це відображено на рисунку 3, а).

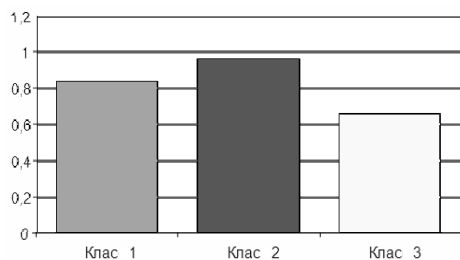


Рисунок 8 - Співпадання силуетів об'єктів на оригінальному та зменшеному на 30% зображенні

У випадку, коли на зображенні наявні декілька чітко відокремлених об'єктів, ланцюжки для модифікації будуть проходити між об'єктами, наскільки це можливо. Таким чином, відстань між об'єктами буде збільшуватись або зменшуватись, залежно від напрямку масштабування, але самі об'єкти будуть лишатись відносно незмінними.

Якщо необхідно надавати модифікації області, що знаходяться всередині об'єкту, то в першу чергу будуть обрані ланцюжки, які потрапляють в однорідні області об'єкту, що масштабується. Це дозволяє зберегти дрібні деталі об'єкту в початковому вигляді.

Щодо рекомендацій по використанню алгоритму, то можна відмітити, що алгоритм найкраще працює на природних зображеннях, тобто на фотознімках, а саме на тих, де зображено одиночний об'єкт на однорідному фоні, або на зображеннях, що містять пейзаж. Алгоритм не рекомендується застосовувати для зображень, що

містять велику кількість значущих об'єктів, які займають більшу частину зображення. Також велике викривлення буде утворюватись на тих зображеннях, де фон містить складний багатокольоровий малюнок.

### Паралельна реалізація модифікованого алгоритму масштабування

З метою скорочення часу масштабування запропонований модифікований алгоритм було реалізовано на платформі CUDA - технології компанії NVidia, що розроблена для створення додатків для масивно-паралельних обчислювальних пристроїв (в першу чергу для GPU починаючи з серії G80) [8, 9]. Розпаралелюванню піддалася функція обчислення енергій: обчислення виконувались на багатьох потоках (нитках), розрахунок енергії кожного пікселя - на окремій нитці незалежно від обчислення енергій інших пікселів. Також була розпаралелена функція обчислення матриці мінімальних ланцюжків: для вертикальних ланцюжків кількість потоків береться рівною кількості пікселів по ширині, кожен потік заповнює відповідні стовбці масиву мінімальними ланцюжками для кожного пікселя (для горизонтальних ланцюжків робота алгоритму буде аналогічною).

Розпаралелювання обчислень дало можливість скоротити їх час порівняно з послідовною реалізацією, що ілюструється графіками на рис.9.

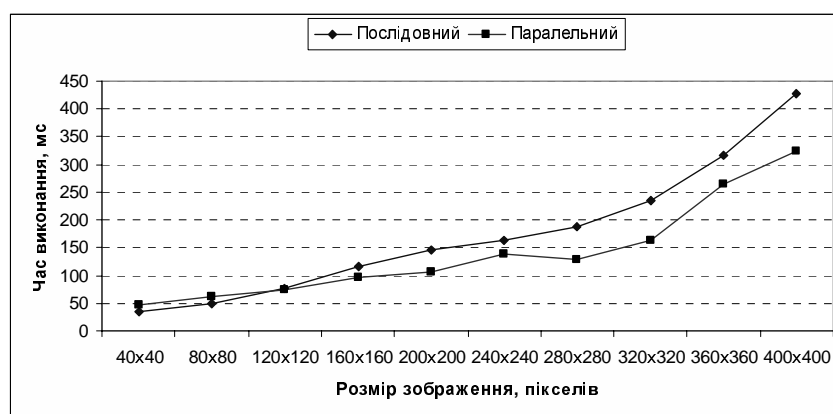


Рисунок 9 – Залежність часу виконання алгоритму від розміру зображення

Як видно з графіка, паралельне обчислення навіть на невеликих зображеннях виконується швидше, ніж послідовне, і різниця між послідовною та паралельною реалізаціями зі збільшенням зображення суттєво зростає на користь паралельної.

### Висновки

У даній роботі була проведена порівняльна оцінка розглянутих алгоритмів масштабування за кількома найбільш важливими характеристиками, виконана класифікація алгоритмів за деякими ознаками, були виявлені їх основні недоліки.

Для оптимізації масштабування був запропонований метод контентно-залежного

масштабування, який змінює зображення в залежності від його вмісту.

Програмна реалізація методу довела, що вміст зображення при масштабуванні зберігається максимально за рахунок видалення/дублювання елементів фону.

Завдяки проведеним випробуванням було проаналізовано роботу запропонованого алгоритму на різних типах та розмірах зображень, проведено аналіз ступеню викривлення зображень при масштабуванні і витрат часу. Запропонована

паралельна реалізація модифікованого алгоритму, орієнтована на використання платформи CUDA, яка дає можливість істотно скоротити час масштабування.

При дослідженні запропонованого алгоритму контентно-залежного масштабування було виявлено, що найменші втрати інформації спостерігаються при масштабуванні природних зображень, фотографій, на яких можна легко відділити об'єкт від фону.

### Список літератури

1. Алгоритмы масштабирования пиксельной графики [Электронный ресурс]. – Электрон. текстові данні (14170 bytes). – Википедия – свободная энциклопедия. – Режим доступу: [http://ru.wikipedia.org/wiki/Алгоритмы\\_масштабирования\\_пиксельной\\_графики](http://ru.wikipedia.org/wiki/Алгоритмы_масштабирования_пиксельной_графики)
2. Atwood J. Better image resizing / J. Atwood // Coding horror: programming and human factors. – Berkeley, 2007. – С.53-59
3. Avidan S. Layer extraction from multiple images containing reflections and transparency / S.Avidan, R.Szeliski, P.Anandan // IEEE Computer Society Conference on Computer Vision and Pattern Recognition. – Hilton Head Island, 2000. – Vol.1. – С.246-253.
4. Shamir A. Non-homogeneous resizing of complex models / A.Sheffer, V.Kraevoy, A.Shamir, D.Cohen-Or // Proceedings of ACM SIGGRAPH. – Asia, 2008. – Vol.27, №5. – С.234-245.
5. General Scaling [Электронный ресурс]. – Электрон. текстові данні (49997 bytes). – Pedagogically-oriented open source site "Leptonica" containing software that is broadly useful for image processing and image analysis applications. – Режим доступу: <http://www.leptonica.com/scaling.html>.
6. Коттер С. Делаем Liquid Resize своими руками / С. Коттер // Habra Digest. – 2009. – №7. – С.6-12.
7. Шапиро Л. Компьютерное зрение / Л. Шапиро, Дж. Стокман. – М.: Бином. Лаборатория знаний, 2006. – 752с.
8. Боресков А.В. Основы CUDA / А.В.Боресков // Издательский отдел факультета ВМиК, МГУ. – М., 2008. – С.53-62.
9. Боресков А.В. Основы работы с технологией CUDA / А.В.Боресков, А.А.Харламов. – М.: ДМК-Пресс, 2010. – 232с.
10. Avidan Sh. Seam carving for content-aware image resizing / Sh.Avidan, A.Shamir // SIGGRAPH 07. – New York, 2007. – С.10.
11. Ватолин Д. Оценка качества методов масштабирования изображений и результаты сравнений разных методов / Д.Ватолин, С.Путилин // International Conference Graphicon. – М., 2003. – С.51-57.

Надійшла до редколегії 30.04.2011

**Н.С. КОСТЮКОВА, С.О. ШЕХОВЦОВ**

Донецкий национальный технический университет

**Контентно – зависимое масштабирование изображений с сохранением значимых объектов**

Проведена сравнительная оценка алгоритмов масштабирования по наиболее важным характеристикам, выполнена классификация алгоритмов, выявлены их основные недостатки. Для уменьшения искажений предложен метод контентно-зависимого масштабирования, изменяющий изображение в зависимости от его содержания

*Масштабирование изображений, контентно-зависимое масштабирование*

**N.S. KOSTYUKOVA, S.O. SHEKHOVTSOV**

Donetsk National Technical University

**Content – based image scaling with meaningful objects saving**

The paper contains comparison and classification of scaling algorithms. For scaling optimization the method of content - based scaling is proposed. This method changes the image depending on its maintenance. Less distortion of image and selection of non- scaling regions are advantages of the offered method.

*Image scaling, content – based scaling*