

АЛГОРИТМИ ПОРІВНЯННЯ ЗОБРАЖЕНЬ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧІ ПОШУКУ ЗА ВМІСТОМ

Здійснено аналіз методів пошуку зображень за вмістом, охарактеризовано їхні недоліки та переваги з огляду на вимоги до сучасних пошукових систем (швидкість пошуку, релевантність, ресурсні затрати, якість отриманих даних та гнучкість пошукових процесів). Акцентовано увагу на перцептивних хеш-алгоритмах (Block Mean Value Based Hash, DCT Based Hash, Radial Variance Based Hash, Marr-Hildreth Operator Based Hash). Наведено офіційні результати тестування швидкості роботи алгоритмів. Визначено найбільш оптимальний алгоритм на сьогодні.

Ключові слова: пошук, вміст, алгоритм, хеш-функція, релевантність.

Вступ

В умовах прискорених темпів інформатизації дедалі більшої ваги набуває проблема пошуку потрібної інформації. Разом зі збільшенням обсягу інформаційних ресурсів постійно зростають вимоги до сучасних пошукових систем. Серед головних із них виділяють швидкість пошуку, релевантність, ресурсні затрати, якість отриманих даних та гнучкість пошукових процесів. Одним із ефективних способів розв'язання проблеми є розробка та впровадження спеціалізованих пошукових систем. Оскільки інформаційні ресурси спеціалізованої пошукової системи класифікують за типами, інформацію можна структурувати. Є два методи пошуку зображень: пошук за метаданими та за вмістом. У цій статті більш детально розглянуто другий підхід.

Пошук зображень за вмістом (англ. Content-based image retrieval (CBIR)), також відомий як запит за вмістом зображення (англ. Query by image content (QBIC)) та отримання візуальної інформації за вмістом (англ. Content based visual information retrieval (CBVIR)) – це одна з технік машинного зору, що вирішує проблему отримання зображень, що є проблемою пошуку цифрових зображень у великих базах даних. Пошук зображень за вмістом є протилежним підходом щодо пошуку зображень за описом. За методами пошуку та обслуговування виділяють чотири типи пошукових систем: системи, що використовують пошукових роботів, системи, керовані людиною, гібридні системи і метасистеми.

В архітектуру пошукової системи зазвичай входять:

- пошуковий робот, який збирає інформацію з сайтів мережі Інтернет або з інших документів;

- індикатор, що забезпечує швидкий пошук за накопиченою інформацією;

- пошуковик – графічний інтерфейс для роботи користувача.

«На основі вмісту» означає, що пошук має аналізувати фактичний вміст зображення, а не метадані, як-от ключові слова, теги та/або опис зображення. Потенційні галузі застосування алгоритмів пошуку за вмістом:

- пошук зображень у мережі Інтернет;
- каталогізація зображень творів мистецтва;
- організація роботи з архівами фотографічних знімків;
- організація каталогів роздрібного продажу товарів;
- медична діагностика захворювань;
- запобігання злочинам і заворушенням;
- застосування у військових цілях;
- питання контролю за поширенням інтелектуальної власності;
- отримання інформації про місце знаходження віддалених зондів і географічне позиціонування;
- контроль за вмістом масивів зображень.

Огляд методів пошуку зображень за вмістом

Виявлення контурів

Є кілька алгоритмів виділення контурів, які працюють добре і надійно. Проте головна проблема цих алгоритмів полягає в тому, що після виділення контурів незрозуміло, що з ними робити. Можна порахувати відносну площу контурів – в ідеалі можна буде сказати, «рябе» це зображення чи «рівне», але на практиці виявляється, що цей критерій працює слабо. Можна спробувати подивитися на Фур'є-образ зображення кордонів – чи немає на зображенні

виражених періодичних контурів. Але все одно подібна інформація дуже слабо характеризує те, як зображення сприйматиметься візуально. Тому цей клас ознак може бути тільки допоміжним.

SIFT-подібна сигнатура

Метод SIFT (scale-invariant feature transform) полягає у виокремленні на зображенні набору опорних точок і використовує взаємне розташування опорних точок як характеристику зображення. Цей метод має кілька варіацій (передовсім пов'язаних з іншими способами вибору опорних точок). Ця група методів допоможе з'ясувати, чи є одне зображення деформованою копією іншого, однак для визначення подібності двох принципово різних, хоч і візуально схожих, зображень метод непридатний (наприклад, дві фотографії одного й того самого кошечки в різних позах матимуть мало спільного для таких методів). Тому в системі пошуку цим методом можна визначити, чи є в базі модифікації шуканого зображення, але не можна знайти подібні в будь-якому іншому сенсі.

Колір

Пошук зображень через порівняння кольірних складових здійснюється за допомогою побудови гістограм їх розподілу. На сьогодні проводяться дослідження з побудови опису, в якому зображення ділиться на регіони за схожими кольоровими характеристиками, і далі враховується їхнє взаємне розташування. Опис зображень кольорами, які є на них, є найпоширенішим, оскільки він не залежить від розміру або орієнтації зображення. Побудова гістограм із подальшим їх порівнянням використовується найчастіше, але не є єдиним способом опису кольірних характеристик.

Текстура

Текстура – це ділянка зображення, де між сусідніми точками можуть бути істотні перепади яскравості і кольору, але яка візуально сприймається як однорідна ділянка (наприклад, трава, поверхня води і т. п.). Є різні методи, які дозволяють визначити межу між текстурами, і хоча вони доволі обчислювально затратні, все ж їх можна використовувати на практиці. Питання радше в іншому: що робити з цією інформацією? Це може дати приблизно такий самий набір ознак, що й контури – уявлення про те, наскільки зображення візуально однорідне – тільки, в деякому сенсі, більш якісно. Отже, ці ознаки теж можуть бути тільки вторинними, хоча їхнє значення дещо більше.

Форма

Опис форми передбачає опис геометричної форми окремих фрагментів зображення. Для її визначення до фрагмента зображення спочатку застосовують сегментацію, або виділення контурів. Є й інші способи, наприклад, фільтрація форм. Часто визначення форми потребує втручання людини, тому що методи на кшталт сегментації складно повністю автоматизувати для широкого класу задач.

Перцептивні хеш-алгоритми

У процесі пошуку зображень за візуальним вмістом порівняння має здійснюватися не за вихідними зображеннями, а за даними, отриманими в результаті попередньої обробки цих зображень, тому що на порівняння безпосередньо самих зображень потрібно надто багато процесорного часу, і в разі використання навіть на мінімальних сховищах, де кількість зображень трохи більше сотні, є абсолютно неефективним. У разі запису даних у рядковому форматі, ефективність методу залежить від алгоритмів, використовуваних для формування рядка. Одним із найбільш ефективних та універсальних є використання перцептивного хеш-алгоритму.

Перцептивні хеш-алгоритми описують клас функцій для генерації порівнянних хешів. Вони використовують різні властивості зображення для побудови індивідуального «відбитка». Надалі ці «відбитки» можна порівнювати один з одним.

Для розв'язання нашої задачі є 4 перцептивних хеш-алгоритми:

- Block Mean Value Based Hash;
- DCT Based Hash;
- Radial Variance Based Hash;
- Marr-Hildreth Operator Based Hash.

Block Mean Value Based Hash

Суть алгоритму полягає у відображенні середнього значення низьких частот. У зображеннях високі частоти забезпечують деталізацію, а низькі – показують структуру. Тому для побудови такої хеш-функції, яка для схожих зображень видаватиме близький хеш, потрібно позбутися високих частот.

Кроки алгоритму

1. Зменшити розмір зображення до 16×16 пікселів.

2. Прибрати колір. Маленьке зображення перекладається в градації сірого, тож хеш зменшується втричі.

3. Знайти середнє. Обчислити середнє значення кольору для всіх 256 пікселів.

4. Побудувати ланцюжок бітів. Для кожного пікселя робиться заміна кольору на 1 або 0 залежно від того, більший він чи менший від середнього.

5. Побудувати хеш. Перевести 256 бітів в одне значення.

6. Отриманий хеш стійкий до масштабування, стискання або розтягування зображення, зміни яскравості, контрасту, маніпуляцій з кольорами. Але головна перевага алгоритму – швидкість роботи. Для порівняння хешів цього типу використовується функція нормованої відстані Хеммінга.

Discrete Cosine Transform Based Hash

Дискретне косинусне перетворення (ДКП, DCT) – одне з ортогональних перетворень, тісно пов'язане з дискретним перетворенням Фур'є (ДПФ) і є гомоморфізмом його векторного простору. ДКП, як і будь-яке Фур'є-орієнтоване перетворення, виражає функцію або сигнал (послідовність зі скінченної кількості точок даних) у вигляді суми синусоїд з різними частотами і амплітудами. ДКП використовує тільки косинусні функції, на відміну від ДПФ, що використовує і косинусні, і синусні функції.

Кроки алгоритму

1. Прибрати колір, щоб позбавитися від непотрібних високих частот.

2. Застосувати медіанний фільтр для зменшення рівня шуму. Суть медіанного фільтра в тому, що зображення розбивається на так звані «вікна», потім кожне вікно замінюється медіаною для сусідніх вікон.

3. Зменшити зображення до розміру 32×32.

4. Застосувати ДКП до зображення.

5. Побудувати хеш.

Головними перевагами такого хешу є стійкість до малих поворотів, розмиття і стиснення зображення, а також швидкість порівняння хешів завдяки їхньому маленькому розміру. Для порівняння хешів цього типу також використовується функція відстані Хеммінга.

Radial Variance Based Hash

Ідея алгоритму полягає в побудові променевого вектора дисперсії (ПВД) на основі перетворення радону. Потім до ПВД застосовується ДКП і обчислюється хеш. Перетворення радону – це інтегральне перетворення функції багатьох змінних уздовж прямої. Воно стійке до обробки зображень за допомогою різних

маніпуляцій (наприклад, стиснення) і геометричних перетворень (наприклад, поворотів).

Кроки алгоритму

1. Прибрати колір, щоб позбавитися від непотрібних високих частот.

2. Розмити зображення за допомогою використання розмиття Гауса для приглушення деяких шумів.

3. Застосувати гамма-корекцію для усунення блякlosti зображення.

4. Побудувати променевий вектор дисперсії.

5. Застосувати ДКП до вектора дисперсії.

6. Побудувати хеш.

Для порівняння хешів цього типу використовується пошук піка взаємно кореляційної функції.

Marr-Hildreth Operator Based Hash

Оператор Марра–Хілдрета дає змогу визначати краї на зображенні. Це край або контур, що відокремлює сусідні частини зображення, які мають порівняно відмінні характеристики відповідно до деяких особливостей. Цими особливостями можуть бути колір або текстура, але найчастіше це сіра градація кольору зображення (яскравість). Результатом визначення меж є карта кордонів. Карта кордонів описує класифікацію меж для кожного пікселя зображення. Якщо межі визначити як різку зміну яскравості, то для їх знаходження можна використовувати похідні або градієнт.

Кроки алгоритму

1. Прибрати колір, щоб позбутися від непотрібних високих частот.

2. Перевести зображення в розмір 128×128.

3. Розмити зображення за допомогою розмиття Гауса.

4. Побудувати оператор Марра–Хілдрета.

5. Застосувати дискретну згортку до LoG (Laplacian Of Gaussian) та до зображення. Отримаємо зображення, на якому чітко видно стрибки яскравості.

6. Перетворити зображення в гістограму. Зображення розбивається на маленькі блоки (5×5), в яких підсумовуються значення яскравостей.

7. Побудувати хеш із гістограми. Гістограма розбивається на блоки 3×3. Для цих блоків розраховується середнє значення яскравості і використовується метод побудови ланцюжка бітів. Виходить бінарний хеш розміром 64 байти.

Порівняння двох хешів забирає небагато часу, хоча розмір хешів відносно великий, оскільки використовується функція нормованої відстані Хеммінга.

У таблиці наведено офіційні результати тестування швидкості роботи алгоритмів.

Таблиця. Результати тестування швидкості роботи алгоритмів

	DTC	MH	Radial	BMB
ЗЧ(с)	911	343	118	58
СЧ(с)	9,7	3,6	1,3	0,6

Висновки

Кожен із розглянутих алгоритмів має свої переваги, але вимоги до сучасних пошукових систем зростають. Оскільки найвагомішою вважають швидкодню алгоритму, то найбільш оптимальним залишається BMB (Block Mean Value Based Hash).

Список використаної літератури

1. Огневой Г. Д. Методы и алгоритмы поиска изображений [Электронный ресурс] / Г. Д. Огневой. – Режим доступа: <http://www.bntu.by/news/67-conference-mido/1580-2014-11-23-09-36-26.html>. – Загл. с экрана.
2. Pears N. 3D Imaging, Analysis and Applications / Nick Pears, Yonghuai Liu, Peter Bunting. – London : Springer, 2012. – 497 p.
3. Zauner C. Implementation and Benchmarking of Perceptual Image Hash Functions / Christoph Zauner. – University of Applied Sciences Hagenberg, 2010. – 107 p.

Artem Karpovych, Andrii Hlybovets

COMPARISON ALGORITHMS FOR SOLVING THE IMAGE RETRIEVAL PROBLEM

In an accelerated pace of a growing amount of information, the problem of finding the right one becomes increasingly important. Along with the expansion of the volume of information resources, the requirements for modern search engines are constantly rising. Among the main ones are the search speed, the relevancy, the resource costs, the quality of the data received, and the search engine flexibility. One of the effective ways to solve the problem is the development and implementation of specialized search engines. Since information resources of a specialized search engine are classified by type, information can be structured. There are two methods for finding images: metadata search and content. In this article, we will take a closer look at the second approach.

In the process of looking for images on the visual content, the comparison process should be done not on the original images but on the data obtained as a result of the pre-processing of these images, because the comparison of images directly with each other takes too much processor time, and if used even on the minimum repositories where the number of images is just over a hundred, it is completely ineffective. In the case of writing the data in a string format, the effectiveness of the method depends on the algorithms used to form the string. One of the most effective and versatile ones is the use of a perceptive hash algorithm. Perceptive hash algorithms describe a class of functions for generating comparable hashes. They use different image properties to build an individual “imprint”. In the future, these “prints” can be compared with each other.

For solving this problem we choose four algorithms: Block Mean Value Based Hash, DCT Based Hash, Radial Variance Based Hash, and Marr-Hildreth Operator Based Hash. Each of the above algorithms has its advantages, but the requirements for the modern search engines systems are only increasing, and the greatest advantage is given by the malicious code of the algorithm, so the most optimal is the Block Mean Value Based Hash (BMB).

Keywords: search, content, algorithm, hash function, relevancy.

Матеріал надійшов 20.04.2018