

УДК 004.94

К.т.н, доцент Замятін Д.С., магістрант Левченко О.С.

Національний технічний університету України
«Київський політехнічний інститут»

АНАЛІЗ SURF-АЛГОРИТМА ДЛЯ ПОШУКУ ЗОБРАЖЕНЬ

Abstract

*Denis Zamyatin, assoc. prof., PhD; Olga Levchenko, student
Analysis of the SURF-algorithm of image searching*

This paper is devoted to the SURF-algorithm of image searching that solves several the most difficult tasks. SURF-algorithm is based on detecting feature points on images and creating their descriptors. The advantages of this algorithm are fast detecting and invariance to scaling and rotating. This algorithm can be used in an engine for image searching.

Вступ

Задача пошуку зображень за їх вмістом полягає у аналізі фактичного вмісту зображення. Термін «вміст» в даному контексті слід розуміти як кольори, форми, текстури, або будь-яку іншу інформацію, яка може бути отримана з самого зображення.

На сьогоднішній день існує декілька алгоритмів, що частково вирішують дану задачу. Найсучаснішими серед них є SURF та SIFT алгоритми [3]. Для подальшої роботи було обрано SURF-алгоритм, оскільки він в декілька разів швидший за SIFT-алгоритм, і, на думку авторів, він більш стійкий до різних перетворень [3].

Термінологія

SURF (Speeded Up Robust Feature) — надійний детектор зображення і дескриптор.

Дескриптор — лексична одиниця, яка служить для опису основного змісту об'єкту.

Матриця Гессе — квадратна матриця елементами якої є часткові похідні деякої функції.

Гессіан — визначник матриці Гессе.

Флуктуація — випадкове відхилення значення величини від середнього значення.

Октава — підмножина фільтрів, що покриває певний інтервал масштабів і має свій характерний крок між розмірами фільтрів.

Мета дослідження та постановка задачі

Задача, що розглядається в даній роботі, полягає в аналізі SURF-алгоритму для пошуку зображень з метою визначення його недоліків та переваг.

Опис алгоритму [1]

SURF-алгоритм вирішує дві задачі — пошук особливих точок [2] на зображенні і створення їх дескрипторів, інваріантних до масштабу і повороту, тобто опис ключової точки буде однаковий, якщо зразок змінить розмір і буде повернутий. Також пошук самих точок має бути інваріантний, щоб об'єкт сцени мав той самий набір ключових точок, що і зразок. Метод шукає особливі точки за допомогою матриці Гессе. Детермінант матриці Гессе (гессіан) досягає екстремуму в точках максимальної зміни градієнта яскравості. Він гарно знаходить плями, кути і границі ліній. Гессіан інваріантний відносно повороту, але не інваріантний відносно масштабу, тому SURF використовує різномасштабні фільтри для знаходження гессіанів.

Розглянемо формування дескрипторів. Після знаходження ключових точок, SURF формує дескриптори. Дескриптор являє собою набір із 64 (або 128) чисел для кожної ключової точки. Ці числа відображають флуктуації градієнта навколо ключової точки. Оскільки ключова точка являє собою максимум гессіана, то це гарантує, що в околицях точки мають бути ділянки з різними градієнтами. Таким чином, забезпечується відмінність дескрипторів для різних ключових точок. Флуктуації градієнта околиць ключової точки рахуються відносно напрямлення градієнта навколо точки вцілому (по всій околиці ключової точки). Таким чином, досягається інваріантність дескриптора відносно повороту. Розмір самої області, на якій обчислюється дескриптор, визначається розміром матриці Гессе, що забезпечує інваріантність відносно масштабу зображення.

Інтегральне представлення зображення

Розглянемо інтегральне представлення зображення, що використовується для ефективного обчислення фільтрів Гессе.

Інтегральне представлення зображення являє собою матрицю, розмірність якої співпадає з розмірністю вихідного зображення, а елементи матриці обчислюються за формулою:

$$I(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j),$$

де $I(i, j)$ — яскравість пікселів вихідного зображення.

Маючи інтегральну матрицю можна швидко вичислити суму яскравостей пікселів довільних прямокутних областей зображення за формулою:

$$SumOfRect(ABCD) = I(A) + I(C) - I(B) - I(D),$$

де ABCD — прямокутник, що нас цікавить.

Обчислення матриці Гессе

Виявлення особливих точок в SURF-алгоритмі базується на обчисленні матриці Гессе. Матриця Гессе для двовимірної функції і її детермінант визначаються наступним чином:

$$H(f(x, y)) = \begin{vmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{vmatrix}$$
$$\det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2$$

Значення гессіана використовується для знаходження локального мінімуму або максимуму яскравості зображення. В цих точках значення гессіана досягає екстремуму.

Теоретично, обчислення матриці Гессе зводиться до знаходження Лапласіана Гаусіан. По суті, елементи матриці Гессе вираховуються як згортка пікселів зображення на фільтри.

Таким чином, в SURF, гессіан вираховується за формулою:

$$\det(H_{approx}) = D_{xx} D_{yy} - (0.9 D_{xy})^2,$$

де D_{xx} , D_{yy} , D_{xy} — згортки по фільтрах. Коефіцієнт 0.9 має теоретичне обґрунтування [1] і коректує наближений характер розрахунків.

Отже, для знаходження особливих точок, SURF проходить по пікселям зображення і шукає максимум гессіана. В методі задається порогове значення гессіана. Якщо вираховане значення пікселя вище порогу — піксель розглядається як кандидат на ключову точку.

Оскільки гессіан являє собою похідну і залежить тільки від перепаду яскравості, а не від абсолютного його значення, то він інваріантний по відношенню до зміщення яскравості зображення. Таким чином, зміна рівня освітленості не впливає на виявлення ключових точок.

Також властивості гессіана такі, що він досягає максимуму як в точці білої плями на чорному фоні, так і чорної плями на білому фоні. Таким чином, метод знаходить і темні, і світлі особливості зображення.

Шкали фільтрів

Оскільки гессіан не інваріантний відносно масштабу, то для одного і того самого пікселя гессіан може змінюватись при зміні масштабу фільтра.

Існує тільки одне рішення цієї проблеми — перебирати різні масштаби фільтрів і по черзі застосовувати їх до даного пікселя.

Враховуючи симетрію і дискретизацію, розмір фільтра гессіана не може приймати довільні значення. Допустимі розміри цього фільтру такі: 9, 15, 21, 27 і т. д. з кроком 6. Проте, на практиці поступово збільшувати розмір фільтру на 6 не ефективно, оскільки для великих масштабів крок в 6 одиниць занадто малий, і фільтри стають надлишковими. Тому, SURF розбиває всю множину фільтрів на октави. Кожна октава покриває певний інтервал масштабів, і має свій характерний розмір фільтра.

Недоліки алгоритму

Незважаючи на те, що SURF використовується для пошуку об'єктів на зображенні, сам алгоритм працює не з об'єктами. SURF ніяк не виділяє об'єкт з фону. Алгоритм розглядає зображення як єдине ціле і шукає особливості цього зображення. При цьому особливості можуть бути як всередині об'єкта, так і на фоні, а також на границях об'єкта і фону. В зв'язку з цим, метод погано працює для об'єктів простої форми і без яскраво вираженої текстури. Всередині таких об'єктів, метод скоріш за все, не виявить особливих точок. Точки можуть бути знайдені або на границі об'єкта з фоном, або взагалі тільки на фоні. Це призведе до того, що об'єкт не буде розпізнаним на іншому зображенні чи на іншому фоні.

Висновки

Вище розглянуто та проаналізовано SURF-алгоритм. Виділені такі переваги алгоритму, як інваріантність до масштабу та повороту зображення, а також розглянуто шляхи досягнення інваріантності. Ще однією перевагою алгоритму є інваріантність до зміщення яскравості зображення. Недоліком алгоритму є те, що він погано розпізнає примітивні зображення, оскільки виділення особливих точок стає неможливим на монотонному зображенні.

Література

1. *H.Bay, A.Ess, T.Tuytelaars,L.Van Gool.* Speeded-Up Robust Features (SURF). // Elsevier. - 2008. - pp. 14.
2. *M. Brown, D. Lowe.* Invariant features from interest point groups.// In BMVC. - 2002. - pp.10.
3. *Luo Juan & Oubong Gwun.* A Comparison of SIFT, PCA-SIFT and SURF. //International Journal of Image Processing (IJIP) Volume 3, Issue 4. - 2010. - pp. 143-152.