

Titel : Lernen von Merkmalsgewichtungen beim inhaltsbasierten Suchen in grossen Bilddatenbanken (Content-Based Image Retrieval)
Autor : Henning Müller
Koautoren : Wolfgang Müller, David Squire, Stéphane Marchand-Maillet, Thierry Pun
E-Mail : henning.mueller@cui.unige.ch
Lehrstuhl : Centre Universitaire d'Informatique
Hochschule : Universität Genf
Betreuer : Prof. Thierry Pun
Art der Arbeit : Doktorarbeit
Fachbereich : Künstliche Intelligenz (FB 1)

Zusammenfassung

Dieser Artikel beschreibt einen Ansatz zum Lernen von Merkmalsgewichtungen beim inhaltsbasierten Suchen (Content-Based Image Retrieval) in Bilddatenbanken. Dazu werden Logfiles des Benutzerverhaltens einer Internet-Demoversion des Systems *Viper* über mehrere Monate analysiert und Gewichtungsfaktoren aus den gewonnenen Daten berechnet, die auf gemeinsam markierten Bildern beruhen. Die Qualität der Resultate vor und nach dem Lernen wird anhand eines Experimentes mit mehreren Benutzern verglichen. Es zeigt sich, dass die Qualität der Antworten durch das Lernen der Gewichtungsfaktoren deutlich gesteigert werden kann.

1. Einleitung

Wegen der Verfügbarkeit von mehr und mehr Bild- und Multimediadaten vor allem durch das World Wide Web (WWW) und die digitale Publikationsbranche nimmt die Wichtigkeit von Werkzeugen zum Verwalten dieser Daten und zum Zugriff auf sie zu. Das bildinhaltsbasierte Suchen (Content-Based Image Retrieval) in grossen Bilddatenbanken ist seit mehreren Jahren ein sehr wichtiger Forschungszweig, der verschiedene kommerzielle Systeme wie z.B. Virage oder IBMs *QBIC* [1] und akademische Systeme wie z.B. Blobworld, IMedia oder *Viper* [2] hervorgebracht hat. Ziel ist dabei die Suche basierend auf Bildinhalten (Farben, Formen, Texturen) und nicht auf textuellen Annotationen durchzuführen.

Bei den meisten Systemen wird allerdings nicht der Versuch unternommen, aus dem Benutzerverhalten zu lernen oder zumindest über mehrere Anfragen zu einem Ziel zu kommen. Es wird stattdessen jeder Anfrageschritt separat betrachtet. Lediglich Systeme wie *PicHunter* [3] versuchen über mehrere Schritte ein konkretes Ziel zu verfolgen und zu finden. Das Verhalten der Benutzer über einen längeren Zeitraum wird bisher noch überhaupt nicht betrachtet.

In anderen Bereichen wie dem ECommerce ist die Auswertung des Benutzerverhaltens, vor allem über einen längeren Zeitraum, schon Normalität und verschiedene Programme für diese Auswertung stehen zur Verfügung [4].

2. Das Viper System

Viper [2] ist ein System zum inhaltsbasierten Suchen, welches vor allem auf Techniken des Text Retrieval (inverted files, frequenzbasierte Gewichtungen, Relevanz-Feedback) beruht. Diese Techniken erlauben es, einen sehr grossen Merkmalsraum aus im Moment über 85.000 möglichen Merkmalen zu benutzen. Die Merkmale sind einfach gehalten und nicht in allen Bildern vorhanden. In der Regel enthalten die Bilder zwischen 800 und 3000 dieser Merkmale. Wir benutzen eine Bilddatenbank des Télévision Suisse Romande mit 2500 sehr verschiedenen Bildern. Für die Auswertung benutzen wir Relevanzbewertungen von 3 Benutzern über 14 Anfragebilder der Datenbank [5].

Als Bildmerkmale benutzt *Viper* globale und lokale Farbinformationen im HSV-Farbraum, sowie globale und lokale Filter für die Detektion von Orientierungen im Bild (Gabor-Filter). Diese Merkmale entsprechen in etwa der menschlichen Perzeption von Farben und Texturen.

Die Merkmale werden nach Häufigkeiten im Bild und nach Häufigkeiten in der gesamten Datenbank gewichtet. Ein Beispielgewichtungsschema ist :

$$\text{Merkmalsrelevanz}_j = \frac{1}{N} \sum_{i=1}^N (tf_{ij} \cdot R_i) \cdot \log^2 \left(\frac{1}{cf_i} \right)$$

$$\text{Ergebnis}_{kq} = \sum_j (tf_{ki} \cdot \text{Merkmalsrelevanz}_j)$$

wobei tf die Häufigkeit eines Merkmals im Bild, cf die Häufigkeit eines Merkmals in der gesamten Kollektion und j eine Merkmalsnummer ist. q ist eine Anfrage mit N Anfragebildern, k ein Ergebnisbild und R die Relevanz eines Eingabebildes im Bereich [-1 ;1].

3. Datenanalyse

Als Grundlage für diese Auswertung benutzen wir die Benutzerlogfiles einer Internetdemonstration des *Viper Systems* [6]. Darin sind alle Aktionen des Benutzers verzeichnet, und mal kann sämtliche Anfrageschritte nachvollziehen. In der Webdemo sind verschiedene Bilddatenbanken verfügbar. Insgesamt haben wir das Benutzerverhalten in der Zeit vom September 1999 bis Januar 2000 untersucht. In dieser Zeit hatten wir 3500 Zugriffe auf unser System, von denen 1700 Suchanfragen waren. Bei den restlichen Zugriffen wurden sich lediglich Bilder angeschaut und zwischen den Datenbanken gewechselt. 786 dieser Anfragen sind Anfragen mit mehreren Eingabebildern. Nur diese Mehrbilderanfragen werden wir zum Lernen benutzen, weil sie genug Informationen für unseren Algorithmus besitzen. Genauere Informationen sind in [7] zu finden.

Queries mit mehreren Bildern können sowohl relevante (positiv) als auch irrelevante (negativ) markierte Bilder enthalten. Unser Algorithmus beruht dabei auf der Tatsache, dass gemeinsam positiv markierte Bilder wichtige Merkmale gemeinsam haben (positives Paar), während Bilder, von denen im selben Anfrageschritt eins positiv und eins negativ markiert sind, nur irrelevante Merkmale gemeinsam haben (negatives Paar).

Die 786 Mehrbilderanfragen mit bis zu 42 Anfragebildern werden zuerst in Bildpaare zerlegt, die noch in positive und negative Bildpaare untergliedert werden. Wir erhalten so insgesamt 31.000 Bildpaare.

Für jedes dieser positiven und negativen Bildpaare untersuchen wir die gemeinsamen Merkmale. Ist ein Merkmal in einem positiven Bildpaar, so bekommt dieses Merkmal eine positive Markierung, während die Gemeinsamkeit in einem negativen Bildpaar zu einer negativen Markierung führt. So erhalten wir für jedes Merkmal eine Anzahl positiver und eine Anzahl negativer Markierungen, aus denen wir einen Faktor für die Wichtigkeit des Merkmals berechnen. Dieser Faktor soll für uns zwischen null und zwei liegen, sodass Merkmale mit lediglich negativen Markierungen nicht mehr berücksichtigt werden, während Merkmale mit nur positiven Gewichtungen ihre Wichtigkeit verdoppeln. Daraus können wir folgenden Faktor berechnen und ihn auch gleich in das in Kapitel 2 beschriebene Gewichtungsschema einfügen :

$$\text{Faktor}_j = 1 + \frac{p_j}{p_j + n_j} - \frac{n_j}{p_j + n_j},$$

$$\text{NeueMerkmalsrelevanz}_j = \text{Faktor}_j \cdot \frac{1}{N} \sum_{i=1}^N (f_{ij} \cdot R_i) \log^2 \left(\frac{1}{cf_i} \right),$$

wobei p die Anzahl der positiven Markierungen und n die Anzahl der negativen Markierungen eines Merkmals j sind.

Diesen Faktor berechnen wir einmal aus allen Anfragen, die mit der Datenbank des *Télévision Suisse Romande* gemacht wurden, und einmal aus den Anfragen aller Datenbanken, um zu sehen, wie gut es möglich ist, aus den Daten einer Bilddatenbank die Faktoren für eine anderen Datenbank zu berechnen.

4. Ergebnisse des Experiments

Für die Auswertung benutzen wir eine Datenbank mit 2500 sehr unterschiedlichen Bildern des *Télévision Suisse Romande* für die wir Relevanzbewertungen von drei Benutzern für insgesamt 14 Anfragen haben. Das Benutzerexperiment ist genauer in [5] beschrieben. Wichtiges Ergebnis des Experiments ist, dass jeder Benutzer sehr unterschiedliche Relevanzbewertungen für die Bilder vornimmt, weswegen wir die Ergebnisse für jeden Benutzer getrennt berechnen und am Ende über alle Benutzer und alle Queries mitteln. Als Auswertung benutzen wir Precision/Recall Graphen, die die Standardmethode im Text Retrieval sind, und auch im Content-Based Image Retrieval sehr häufig benutzt werden.

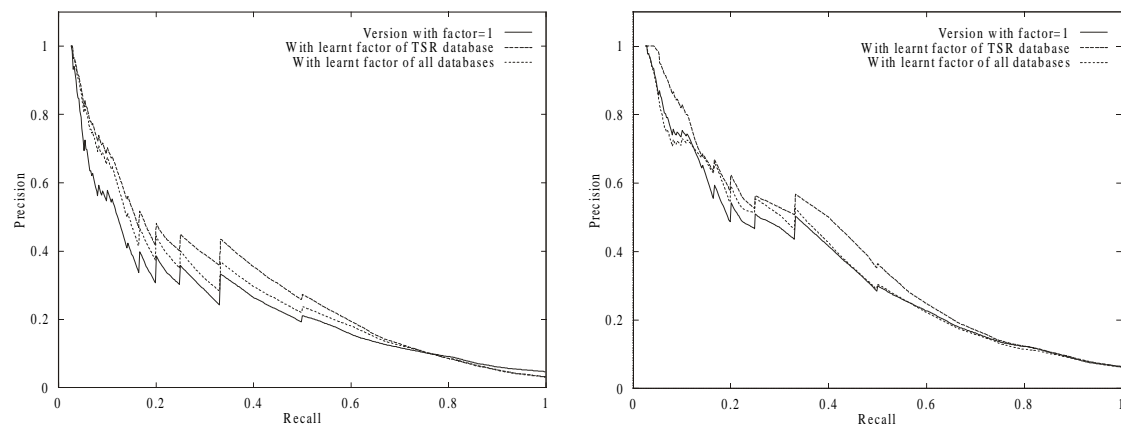


Abbildung 1 : Precision/Recall Graphen mit und ohne Feedback.

Für die Auswertung berechnen wir unseren Faktor einmal aus sämtlichen Anfragen an das System und einmal nur aus den Anfragen an die Datenbank des Télévision Suisse Romande. Zusätzlich vergleichen wir die Ergebnisse mit dem System vor der Lernphase.

Die Grafiken zeigen die Precision/Recall Graphen in einem ersten Querschnitt, sowie die Ergebnisse nach Relevanz-Feedback.

In Abbildung 1 können wir gut sehen, dass die Ergebnisse sowohl mit als auch ohne Feedback durch den zusätzlichen Faktor deutlich verbessert werden. Vor allem im ersten Querschnitt ist das Ergebnis um bis zu 20% besser. Die besten Ergebnisse werden erzielt, wenn der Faktor aus den Anfragen derselben Datenbank berechnet wird. Das ist naheliegend, da unterschiedliche Datenbanken unterschiedliche Bereiche des Merkmalsraums benutzen. Die Ergebnisse mit Feedback sind nur etwa 10% besser mit dem gelernten Faktor, was aber immer noch sehr deutlich sichtbar ist. Vor allem der wichtige Anfang eines Precision/Recall Graphen wird deutlich verbessert, was bedeutet, dass vor allem die auf dem Bildschirm sichtbaren ersten Antwortbilder besser sind.

5. Zusammenfassung und Ausblick

Wir haben in den Ergebnissen des Experiments gesehen, dass die Resultate durch das Lernen von Merkmalsgewichtungen deutlich verbessert werden können, vor allem wenn mit Anfragen aus derselben Datenbank gelernt wird.

Die Vermutung liegt Nahe, dass ein Lernen auf Benutzerbene noch effizienter sein kann, da ein und derselbe Benutzer vermutlich konsistentere Antworten gibt. Ausserdem geben echte Benutzer sicherlich qualitativ bessere Antworten als zufällige Testnutzer im Internet, die das Programm nur einmal ausprobieren.

Zudem werden wir mit unterschiedlichen Formeln für die Berechnung des Faktors experimentieren, denn je nach Datenbank oder Benutzerdaten könnten unterschiedliche Faktoren mit stärkerer oder geringerer Gewichtung zu besseren Ergebnissen führen. Vor allem bei ähnlichen oder wiederkehrenden Suchanfragen wird eine Verstärkung des maximalen Faktors von zwei auf z.B. 4 bessere Resultate bringen.

Referenzen

- [1] QBIC – IBMs Query By Image Content. <http://www.qbic.almaden.ibm.com/~qbic/>, 1999
- [2] D. M. Squire, W. Müller, H. Müller and J. Raki. Content-based query of image databases, inspirations from text retrieval : inverted files, frequency-based weights and relevance feedback. In *The 11th Scandinavian Conference on Image Analysis (SCIA '99)*, pages 143-149, Kangerlussuaq, Greenland, 1999.
- [3] I. J. Cox, M. L. Miller, S. M. Omohundro and P. N. Yianilos. Target Testing and the PicHunter Bayesian multimedia retrieval system. In *Advances in Digital Libraries (ADL '96)*, pages 66-75, Library of Congress, Washington D.C., 1996.
- [4] B. Berendt and M. Spiliopoulou. Analysis of navigation behaviour in web sites integrating multiple information systems. *VLDB Journal*. Special Issue on Databases and the web, 2000
- [5] H. Müller, D. M. Squire, W. Müller and T. Pun. Efficient methods for content-based image retrieval with inverted files. In s. Panchanathan, S.-F. Chang and C.-C. Kuo, *Multimedia Storage and Archiving Systems IV (VV02)*, volume 3846 of SPIE Proceedings, Boston, Massachusetts, USA, 1999.
- [6] Viper Webdemonstration : <http://viper.unige.ch/>, 2000
- [7] H. Müller, W. Müller, S. Marchand-Maillet, D. Squire, T. Pun. Learning Feature Weights from User Behavior in Content-Based Image Retrieval. *Workshop on Multimedia Data Mining* at the SIKDD, Boston, Massachusetts, USA, 2000.