Яндекс на РОМИП 2010: Поиск похожих изображений и дубликатов

© Слесарев А.В.[1,2], Мучник И.Б.[3], Михалев Д.К.[1],

Крайнов А.Г.[1], Котляров Д.И.[1], Беляев Д.В.[1]

[1] Яндекс, [2] МФТИ, [3] Университет Ратгерса
{slesarev, krainov, dmikhalev, dimko, dvbelyaev}@yandex-team.ru,
muchnikilya@yahoo.com

Аннотация

В статье описаны подходы используемые коллективом разработчиков Яндекса для выполнения заданий РОМИП`2010 по поиску похожих изображений и поиску дубликатов.

1. Дорожка поиска похожих

1.1. Введение

Существуют огромные коллекции изображений из интернета. Эффективные методы поиска в таких больших коллекциях должны максимально лаконично описывать каждое изображение и иметь быструю процедуру сравнения пары изображений. На данную работу нас вдохновила статья[2], в которой показано, что человеческое восприятие позволяет с хорошей точностью понимать содержание изображений с маленьким разрешением. Для серых изображений, человеку нужно разрешение около 64x64 пикселя, чтобы хорошо распознавать объекты и окружение. В случае цветных картинок, человеку достаточно разрешения 32x32, чтобы достичь точности распознавания сцен превышающей 80% .

На больших коллекциях простой метод позволяет достичь хороших результатов, по причине того, что коллекция настолько

большая, что похожие изображения обязательно найдутся. Целью данной работы было оценить качество работы методов использующих уменьшенные копии изображений на небольшой по сравнению с веб-масштабами коллекции РОМИП 2010 похожих изображений.

1.2. Описание алгоритма TinyImages

Изначально генерируется уменьшенная копия изображения разрешением 32x32 пикселя. Каждый пикселю приписывается цвет в пространстве RGB. В качестве расстояния используется сумма квадратов попиксельной разности двух изображений: I_1 и I_2:

$$D^2_{\text{ssd}} = \sum_{x,y,c} (I_1(x,y,c) - I_2(x,y,c))^2$$

Здесь x, y – координаты, а c – номер цвета.

1.3. Описание метода Hist

Мы также проанализировали работу методов, в основе которых лежит идея разбивать изображение на небольшое число прямоугольных областей и считать гистограммы для каждой области. К сожалению, данные методы не показали достойных результатов, поэтому подробно описывать представленный вариант мы не будем.

1.4. Результаты

Всего оценивалось 250 запросов, которые были отобраны случайно. В пулы включались по 20 первых результатов в каждом прогоне, при этом изображение образец пропускалось (то есть в таком случае брали 21-й элемент). Каждое задание оценивало 3 асессора.

Использовалось три способа сведения оценок:

- or - хотя бы один из асессоров признал ответ релевантным
- and - все асессоры признали ответ релевантным
- vote - большинство асессоров признало ответ релевантным

Каждому изображению асессору ставили одну из трех оценок:

- очень похожи.(Strong)

- отдаленно близки(Weak)
- непохожи

С некоторой долей условности можно определить а) **очень похожие** картинки как те, которые обладают достаточным визуальным и смысловым сходством; б) **отдаленно близкие** – как обладающие смысловым и некоторым визуальным сходством; в) **непохожие** – для которых отсутствует как визуальное, так и смысловое сходство. Мы приводим только наиболее интересные на наш взгляд метрикам для схем AND-strong (Рис. 1) и AND-weak (Рис. 2).

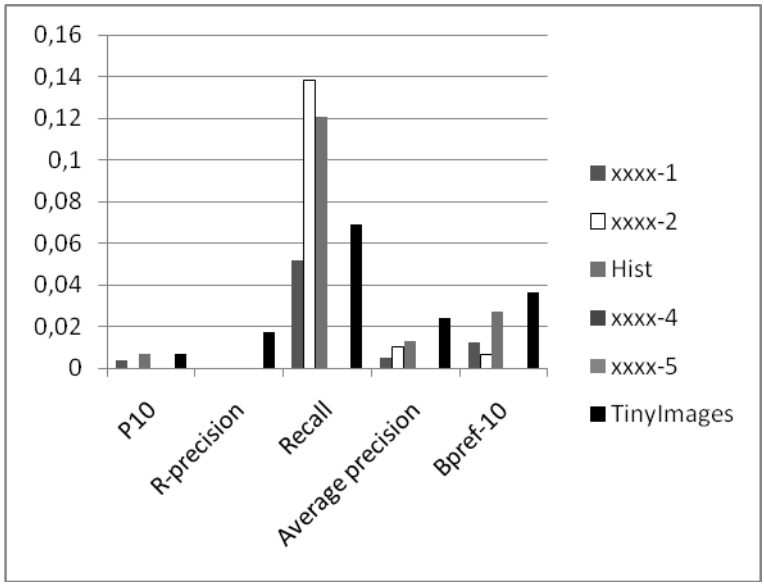

Рис. 1 Результаты участников дорожки поиска изображений по содержанию, оценка AND-strong.

Видно, что по схеме weak, лидерами являются xxxx-1 и TinyImages алгоритм. Результаты по остальным метрикам и способам оценки weak, не включенные в данный отчет, демонстрируют аналогичное поведение сравниваемых систем. Результаты strong очень низкие для всех систем, поэтому достаточно сложно давать оценки. Тем не менее, по всем метрикам кроме Recall метод TinyImages занимает лидирующую позицию.

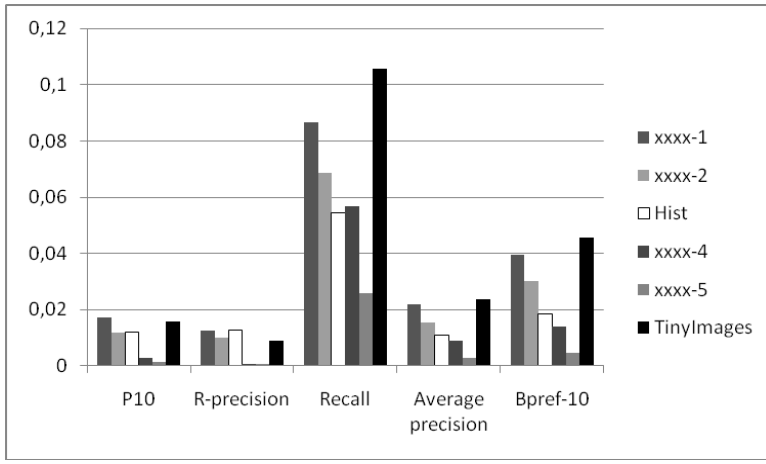

Рис. 2. Результаты участников дорожки поиска изображений по содержанию, оценка AND-weak.

1.5. Выводы

Проведенные эксперименты показали, что, несмотря на свою простоту, описанный метод показывает хорошие результаты на коллекции РОМИП`2010. Предлагается использовать его как «baseline» при оценке качества различных методов поиска похожих.

2. Дорожка поиска дубликатов

2.1. Описание метода

Предварительный поиск дубликатов производится по нескольким сигнатурам (векторам размерности 64 и 128), окончательная проверка производится попиксельным сравнением.

На первом шаге, независимо от пропорций изображение сжимается до размера 20x20 пикселей и приводиться к серому. В качестве первой сигнатуры используется цетральная часть изображения размером 16x16 пикселей. В качестве второй сигнатуры используются дескрипторы интересных точек, координатами которых являются экстремумами преобразования Difference of Gaussian (DoG) [3].

Вводится эмпирический порог, ниже которого сигнатуры считаются «близкими». Используя «близкие» сигнатуры, мы получаем группы кандидатов в дубликаты. Анализируя координаты

«близких» сигнатур на паре изображений, выделяется область пересечения изображений. Для области пересечения вычисляется попиксельная разность и разность карт градиентов. И если достаточный процент этих разностей меньше порогов, то пара изображений считается дубликатами.

Ставим в соответствие каждой картинке число – количество сигнатур других картинок, близких к сигнатурам этой картинки. Картинки сортируются по убыванию счётчика. Для каждой неотмеченной картинки ищутся близкие по сигнатурам, для найденной картинки и картинки запроса производится поиск (по сигнатурам) общей области. Сравнивается область пересечения изображений, по этому сравнению делается вывод об близости изображений, если порог пройден, то картинки считаются полудубликатами и они помечаются. После обработки всех найденных картинок поисковая картинка также помечается. Процесс продолжается пока все картинки не будут помечены. Для уточнения результата (и увеличения кластеров полудубликатов) этот этап повторяется несколько раз с выкидыванием подклеенных картинок (с сохранением привязки) и снижением порогов похожести. После чего для каждой картинки определяем наиболее близкую к ней картинку – «центр» кластера.

В настоящее время оптимизированная версия данного метода позволяет формировать группы дубликатов для базы картиночного поиска, состоящей из 2 млрд. изображений.

2.2. Анализ результатов

График с результатами по полноте и точности можно увидеть на Рис. 3.

3. Заключение

Мы используем РОМИП для проверки гипотез и настройки различных параметров алгоритмов. Несмотря на то, что зачастую постановка заданий семинара отличается от наших задач, опыт участия позволяет развивать методы и лучше понимать границы их применения.

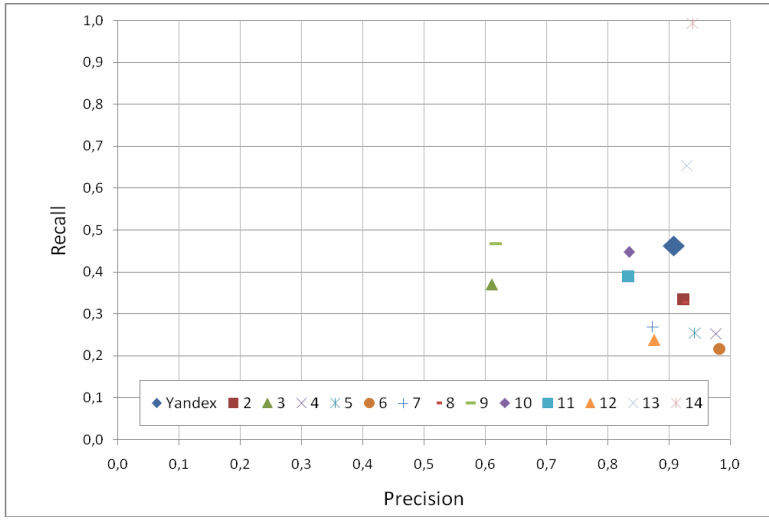

Рис. 3. Положение участников дорожки дубликатов на графике полнота-точность.

Литература

[1] ROMIP Web site, 2010. http://romip.ru

[2] Antonio Torralba, Rob Fergus and William T. Freeman 80 million tiny images: a large dataset for non-parametric object and scene recognition. In IEEE PAMI, 2008

[3] Web site. http://fourier.eng.hmc.edu/e161/lectures/gradient/node11.html

Yandex at ROMIP 2010: CBIR approach for huge databases.

Slesarev A.V., Muchnik I.B., Mikhalev D.K., Krainov A.G., Kotlyarov D.I., Belyaev D.V.

In this paper we describe CBIR approaches for similar image search and image duplicates clustering, used by Yandex at ROMIP`2010.