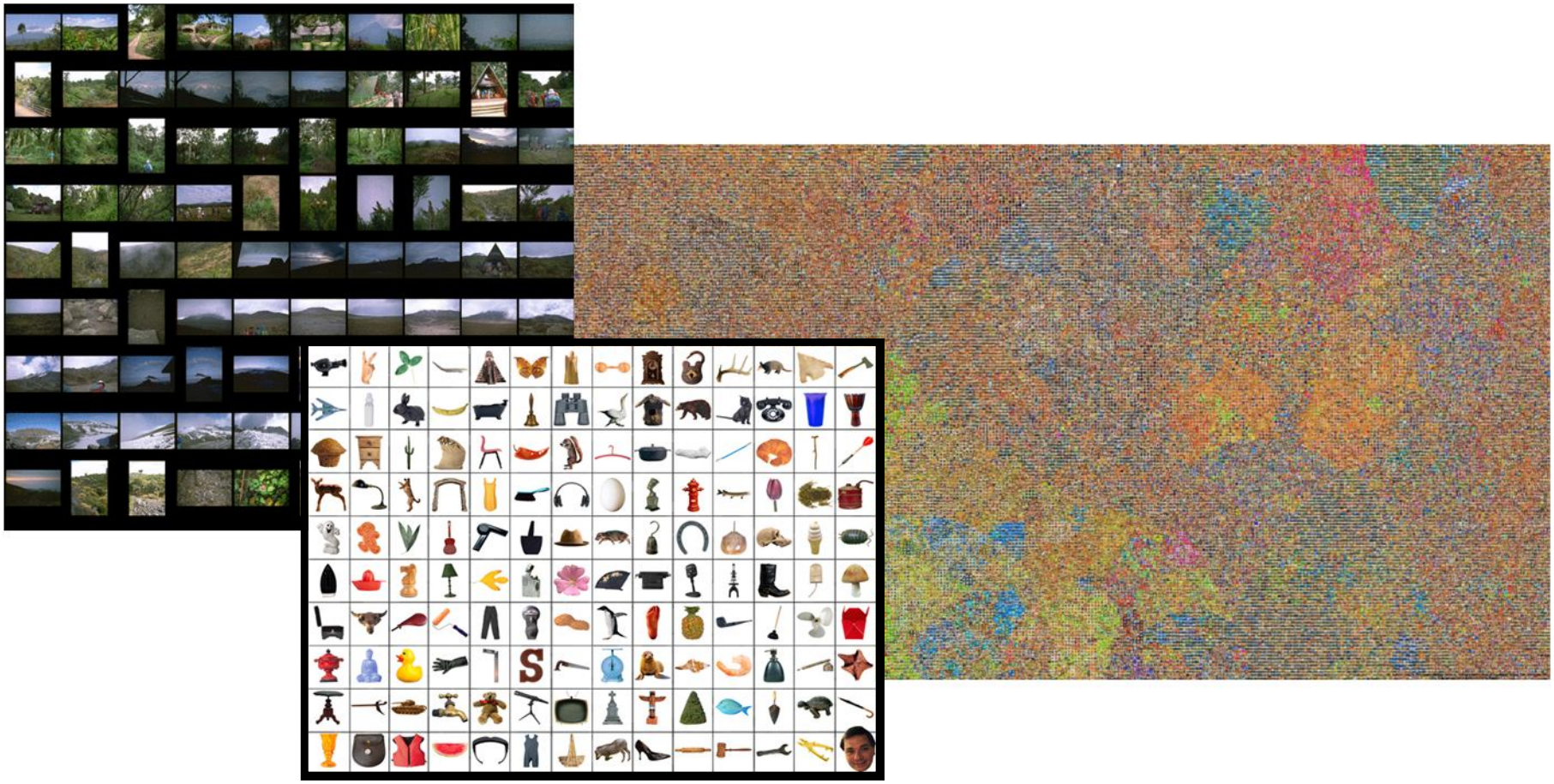
Поиск изображений по содержанию

Microsoft Research



Many slides adopted from Cordelia Schmid, Li Fei-Fei, Rob Fergus, Antonio Torralba

Общая информация

Microsoft Research

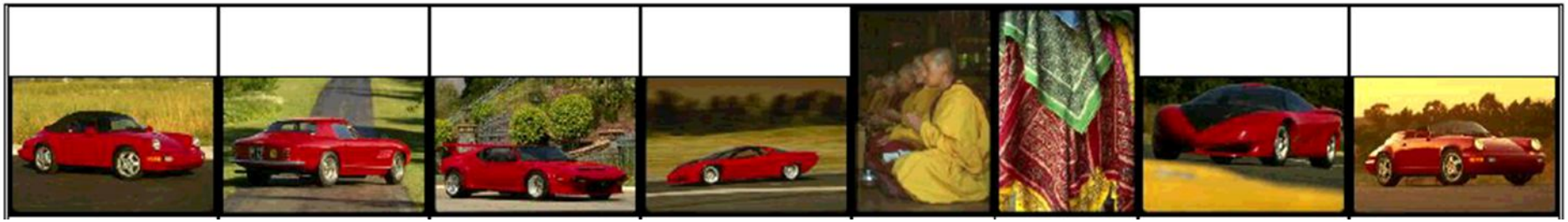
Этот курс подготовлен и читается при поддержке Microsoft Research

- Страница курса http://courses.graphicon.ru/main/vision

Microsoft Research

Задача

- Content-based image retrieval
- Поиск изображений в базе изображений по «какому-то описанию содержимого»
- Задача похоже на классификацию/поиск объектов, но фокусируется в основном на масштабировании и взаимодействии с пользователем

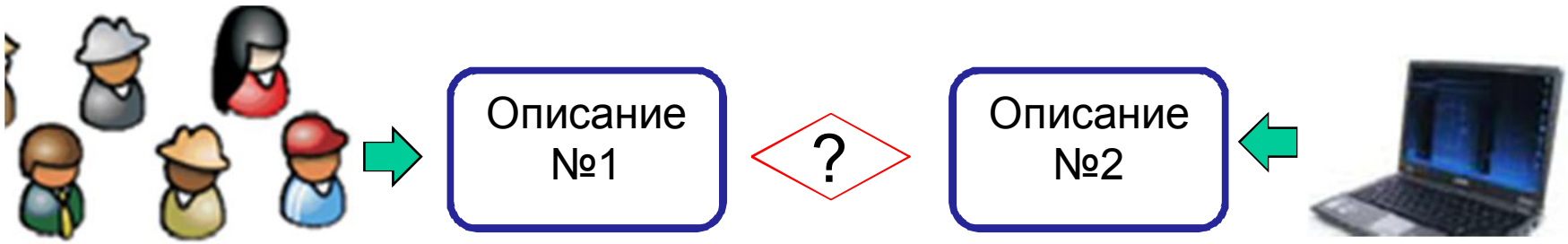


R. Datta, D. Joshi, J. Li, and J. Z.Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys, 2008.*

Semantic Gap

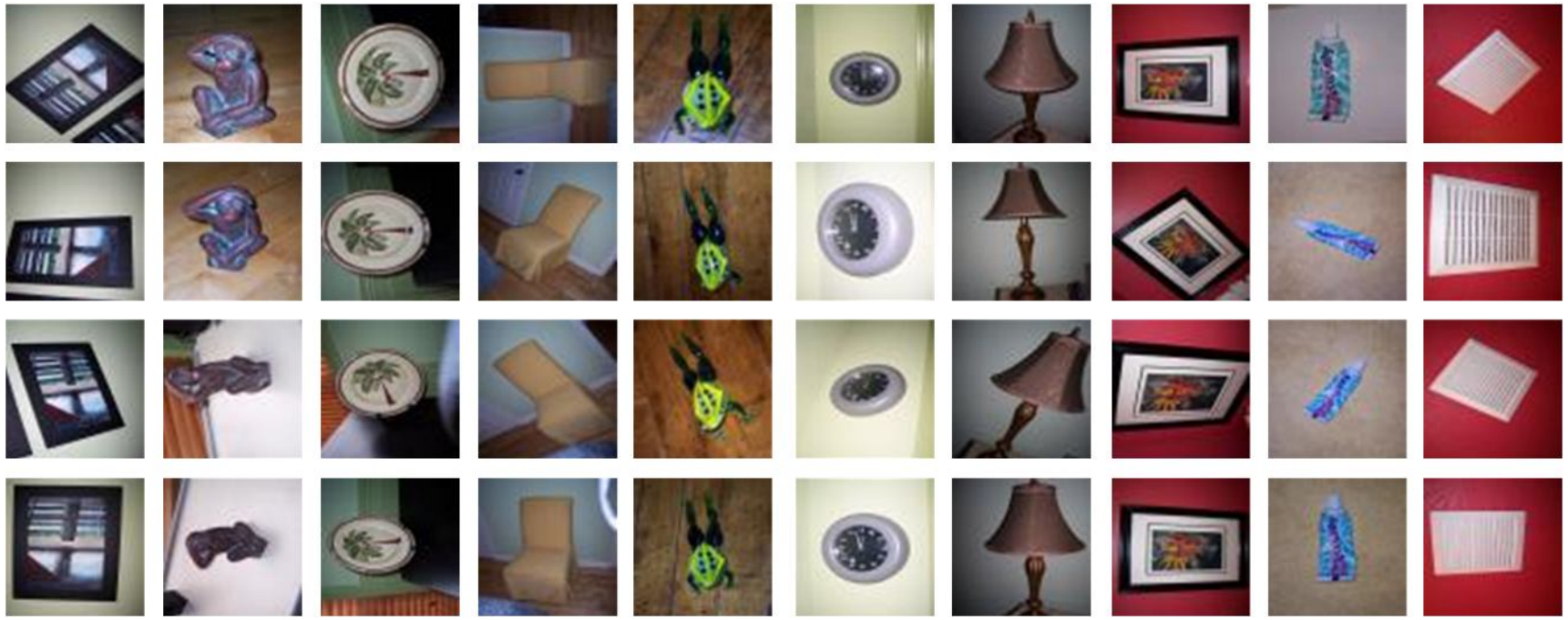
- «Семантический разрыв» – несовпадение информации, которую можно извлечь из визуальных данных, и интерпретацией тех же самых данных со стороны пользователя

- Что значит «похожее изображение»?



Что значит похожее?

1. «Near-duplicates» - изображения фактически одного и того же объекта



Что значит похожее?

2. Похожие по конфигурации сцены, хотя могут быть и разные по назначению

Кухня

Ресепшн

Бар

Автобус

Самолет

Зал

Что значит похожее?

3. «Category-level scene classification» - изображения из одного класса сцен

Пример – банкетный зал.

Что значит похожее?

4. «Category-level classification» - изображения из одного класса объектов

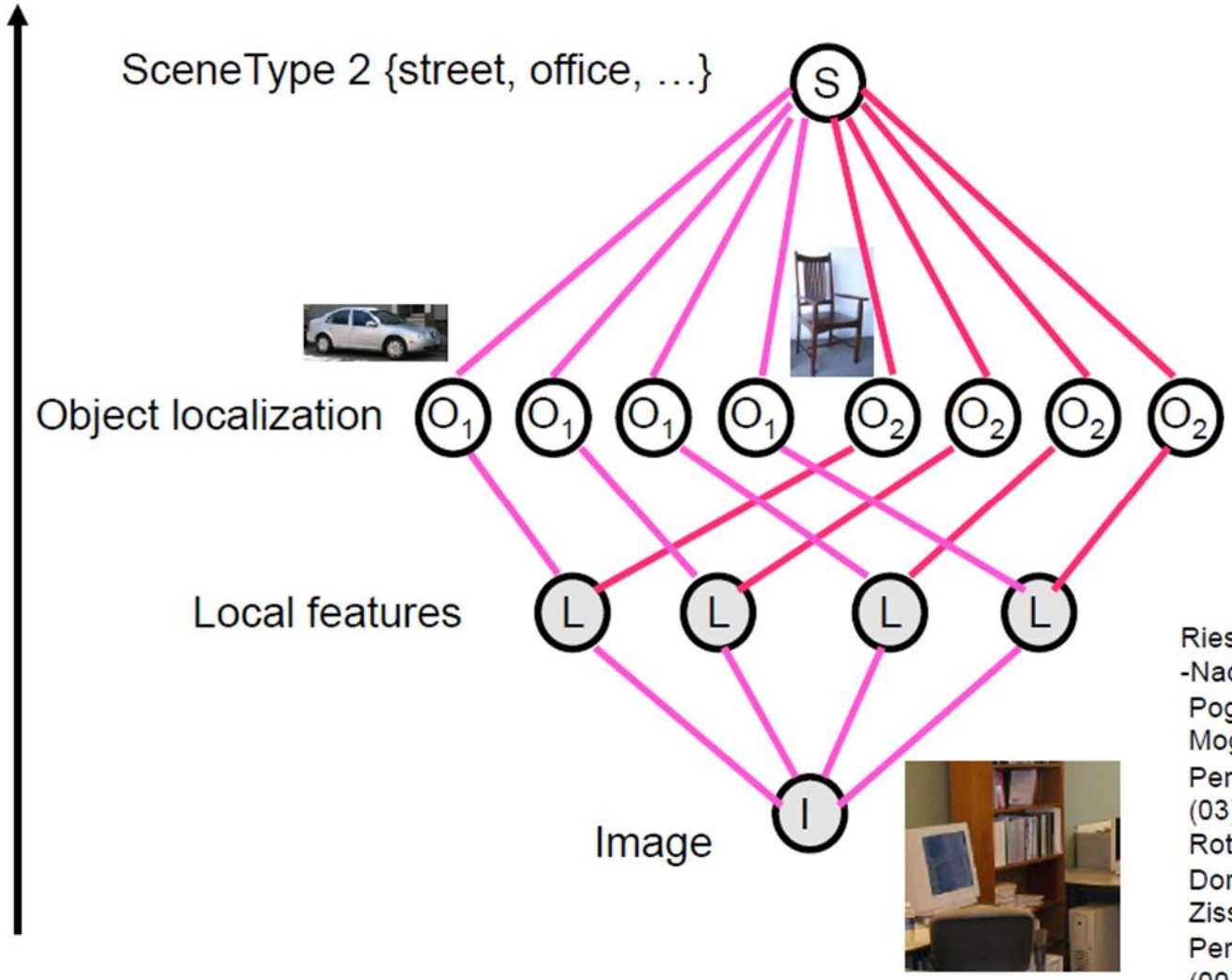
Например, 256 классов из базы Caltech 256

Microsoft Research

Анализ задачи

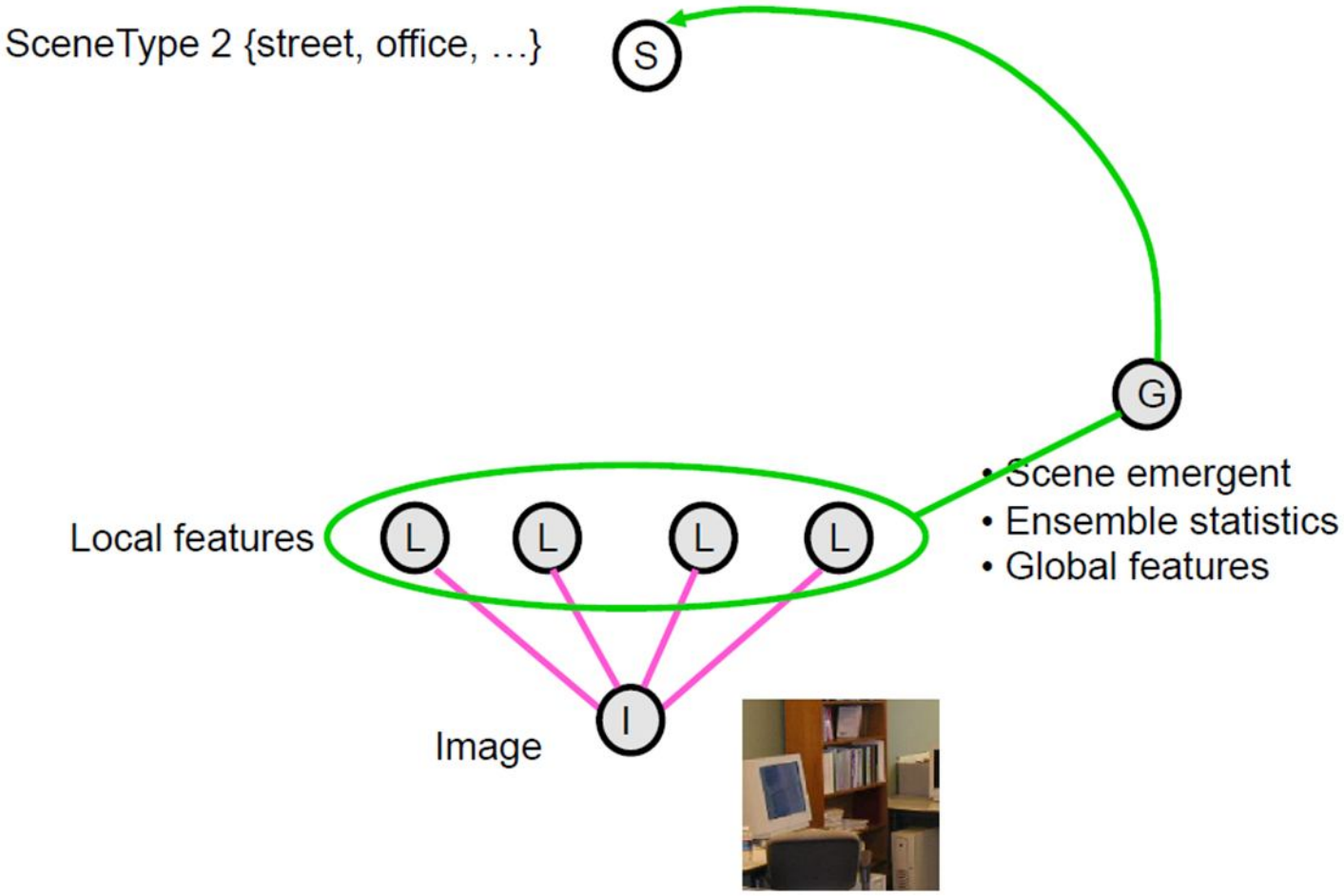
- Все четыре постановки задачи существенно отличаются друг от друга!
- Нам потребуются разные признаки для каждой постановки
- Что ещё?
  - Изображений может быть очень много (в перспективе – миллиарды)
  - Нам нужно описать коллекцию как можно компактнее

Представление сцены



Riesenhuber & Poggio (99); Vidal -Naquet & Ullman (03); Serre & Poggio, (05); Agarwal & Roth, (02), Moghaddam, Pentland (97), Turk, Pentland (91),Vidal-Naquet, Ullman, (03) Heisele, et al, (01), Agarwal & Roth, (02), Kremp, Geman, Amit (02), Dorko, Schmid, (03) Fergus, Perona, Zisserman (03), Fei Fei, Fergus, Perona, (03), Schneiderman, Kanade (00), Lowe (99)

Стандартный подход



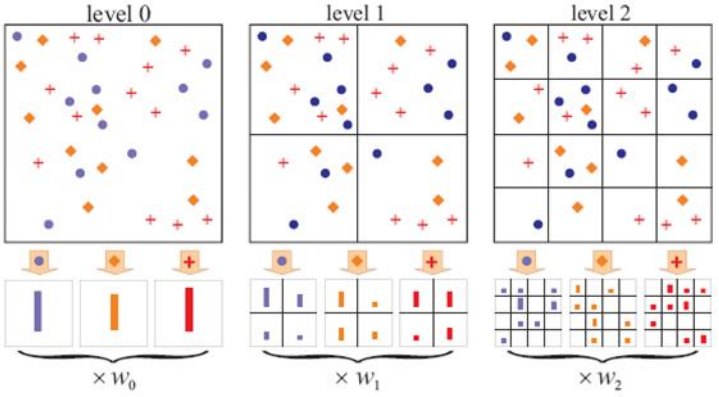
- Построение описания сцены по множеству низкоуровневых локальных особенностей

Глобальные дескрипторы

Гистограммы цвета, яркости (Давно)

GIST (2003)

Мешок слов (2003)



Мешок слов по пространству (2006)

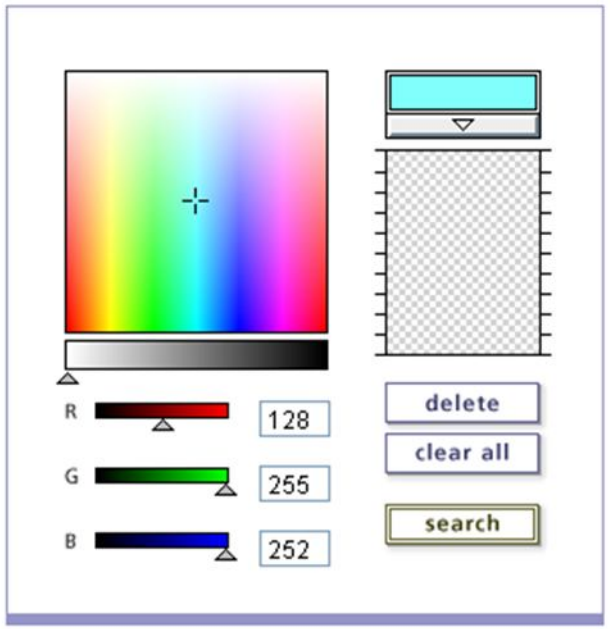
Microsoft Research

QBIC (1995)

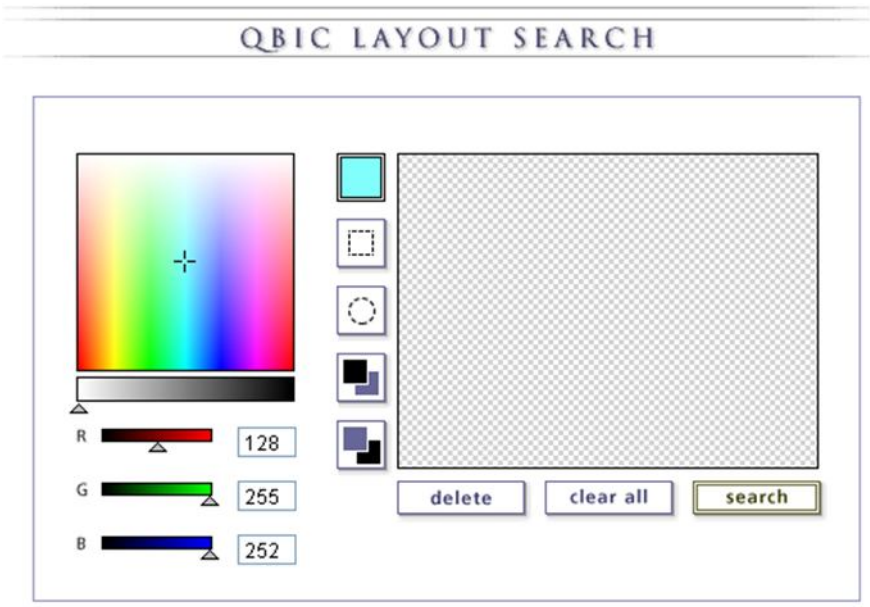
- Query By Image Content
- Вычисляет набор признаков
  - Цветовая гистограмма
  - Набор объектов и их признаков
    - Бинарная маска для описания объектов
    - Ручная или автоматическая сегментация
      - » Выделение контрастных объектов на фоне (музейные экспонаты)
      - » “Snakes”, “заливка» для автоматизированной разметки
    - Признаки формы объектов для распознавания
      - » Площадь, периметр, и т.д.
- ~10000 изображений в базе

M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: the QBIC system. *IEEE Computer, 28(9):23–32, 1995.*

QBIC: Интерфейс

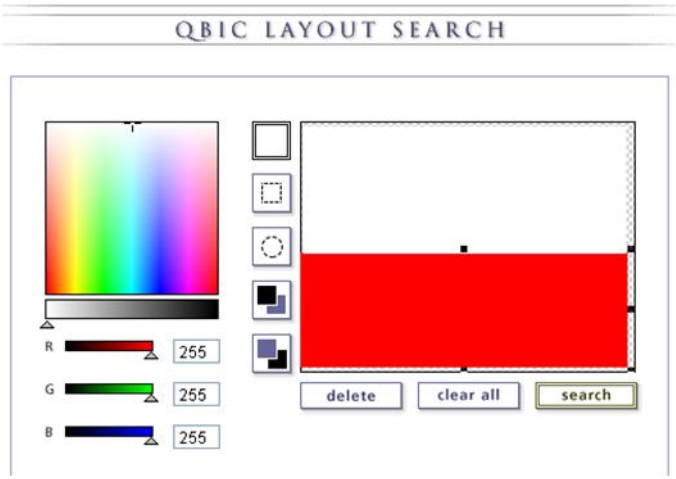


Гистограмма



Пространственное распределение цветов

QBIC: Пример использования



1) **Vase of Flowers**

Huijsum, Jan van Early 18th century

2) **Seascape with Venice in the Distance**

Cottet, Charles Circa 1896

3) **Boats on a Sea Shore**

Goyen, Jan Jozefsz van 1641

4) **Avenue in a Park**

Watteau, Antoine Circa 1715

5) **Bird Perching on a Rose Twig**

UNKNOWN 18th century

6) **Old Woman with a Spindle**

Watteau, Antoine 1710s

7) **Interiors of the New Hermitage. The Room of Russian Sculpture**

Premazzi, Luigi 1854

8) **Allegory of George I, King of England**

Vanloo, Carle (Charles-Andre) 1736

GIST & BOW

- GIST
  - Конфигурация сцены

- Bag of words
  - Одна и та же сцена с разных ракурсов

Microsoft Research

Размеры дескрипторов

- GIST
  - Решетка 4x4 * 8 ориентаций * 4 масштаба = 512 параметров
  - 16384 бита (при 4 байтах на параметр)

- Bag of word
  - Количество слов в словаре (От 64 до 1000000)
  - По пространству – увеличение в 5-18 раз
  - Признак высокой размерности (До 18M параметров!)

Microsoft Research

Поиск на основе «мешка слов»

- Раз «Мешок слов» так хорошо работает для многих задач, построим алгоритм поиска на его базе

- Построение индекса для коллекции изображений:
  - Извлечение особенностей
  - Обучение словаря (кластеризация)
  - Квантование особенностей по словарю (сопоставление)
  - Построение гистограммы частот слов
  - Запись всех «мешков» в каком-то виде

- Поиск изображения
  - Извлечение особенностей
  - Квантование особенностей по словарю (сопоставление)
  - Построение гистограммы частот слов
  - Сравнение гистограммы со всеми из индекса

«Мешок слов»

Microsoft Research

Dataset	# images	# features	Size of descriptors
5K	5,062	16,334,970	1.9 GB
100K	99,782	277,770,833	33.1 GB
1M	1,040,801	1,186,469,709	141.4 GB
Total	1,145,645	1,480,575,512	176.4 GB

- Применим стандартный подход BoW к большой коллекции (5к, 100к, 1М изображений)
- Проблемы:
  - Как построить словарь, если нужно кластеризовать очень большой объём данных? (20М+ векторов для 5К изображений)
  - Как ускорить сопоставление слова словарю (кватнование) при больших размерах?
  - Как хранить индекс изображения?

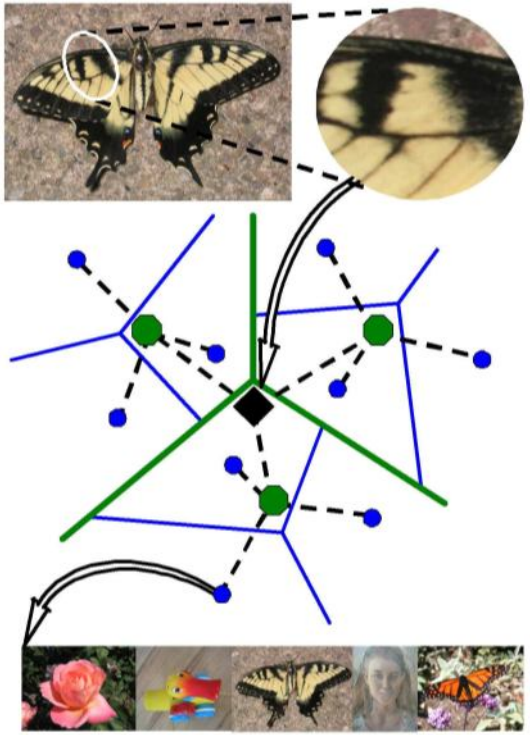
Приближенная кластеризация

Microsoft Research

- Hierarchical k-means (HKM)
- Approximate k-means (AKM)

Hierarchical k-means (HKM)

- «Словарное дерево»
- Иерархическое разбиение
  - Кластеризуем всё на K кластеров (K=10)
  - Затем данные в каждом кластере снова на K кластеров
- Пример:
  - Глубина 6 даёт 1M листьев



D. Nist´er and H. Stew´enius. Scalable recognition with a vocabulary tree. In Proc. CVPR, 2006.

Approximate k-means (AKM)

Microsoft Research

- Алгоритм
  - Лес из 8 рандомизированных k-d деревьев
  - Параметр (координата) разбиения выбирается случайно из набора с наибольшим разбросом
  - Порог разбиения выбирается случайно недалеко от медианы
- Такое разбиение позволяет уменьшить эффекты квантизации
- Сложность каждого этапа k-средних падает с O(NK) до O(Nlog(K))

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” CVPR, 2007.

Оценка

Microsoft Research

Clustering parameters		mAP	
# of descr.	Voc. size	k-means	AKM
800K	10K	0.355	0.358
1M	20K	0.384	0.385
5M	50K	0.464	0.453
16.7M	1M		0.618

- Сравнение AKM с обычным K-means показывает небольшое падение точности (1%)
- На больших размерах выборки обычным K-means слишком долго считать

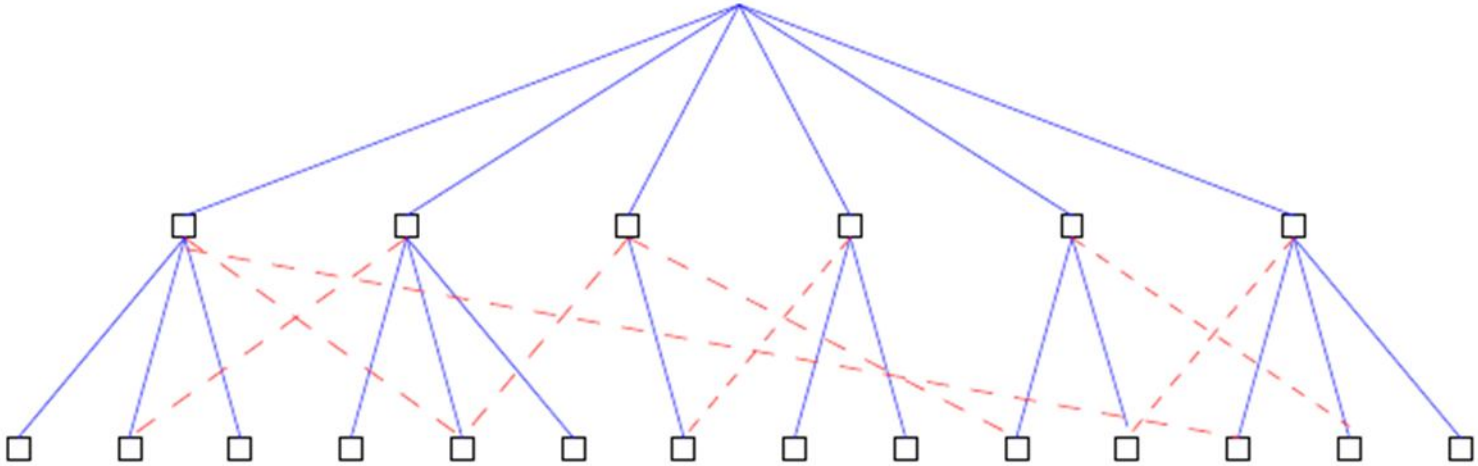
Microsoft Research

Сравнение

Method	Dataset	mAP	
		Bag-of-words	Spatial
(a) HKM-1	5K	0.439	0.469
(b) HKM-2	5K	0.418	
(c) HKM-3	5K	0.372	
(d) HKM-4	5K	0.353	
(e) AKM	5K	0.618	0.647
(f) AKM	5K+100K	0.490	0.541
(g) AKM	5K+100K+1M	0.393	0.465

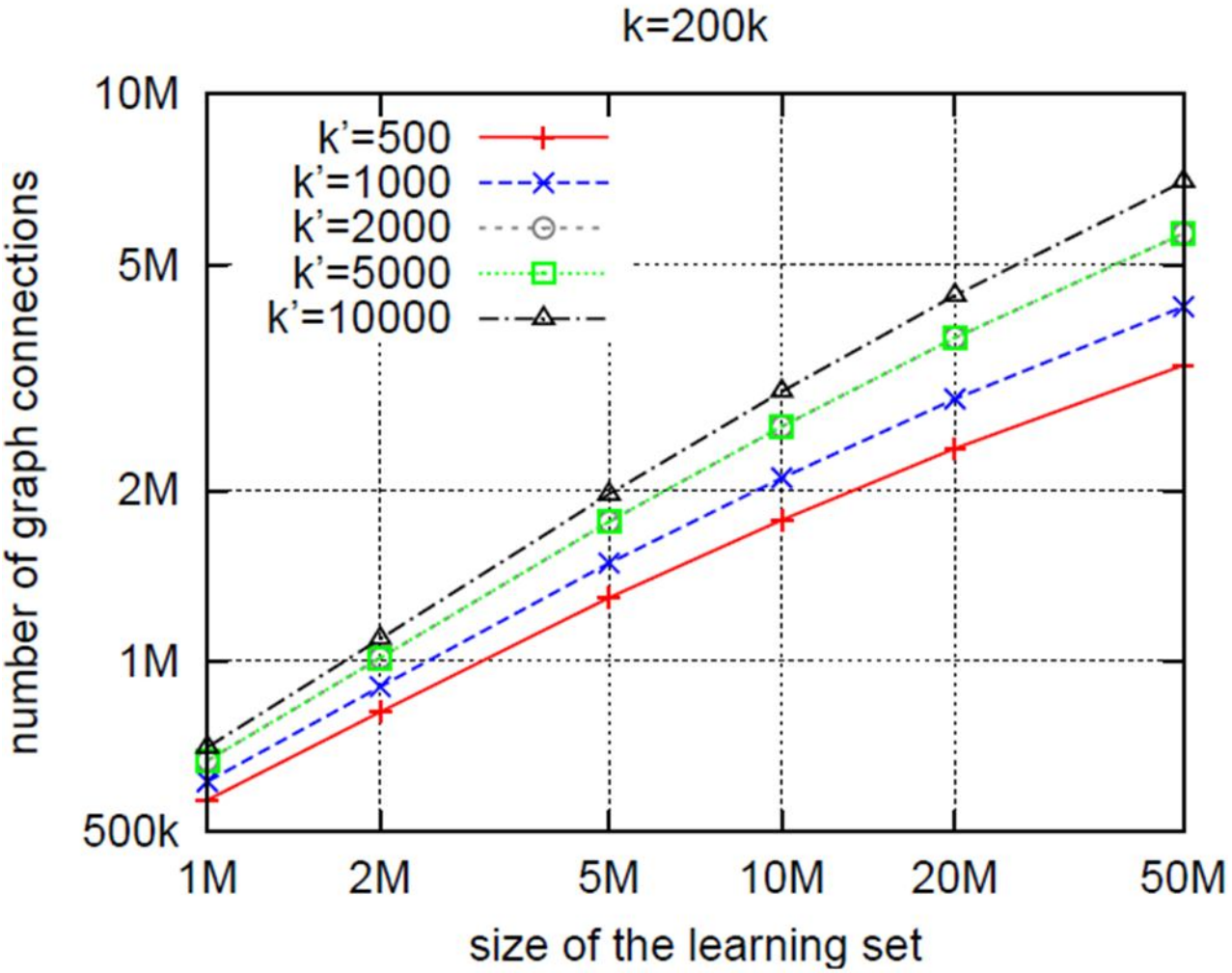
- Выводы:
  - AKM превосходит по точности HKM
  - С ростом размера выборки точность сильно падает

Ускорение квантования



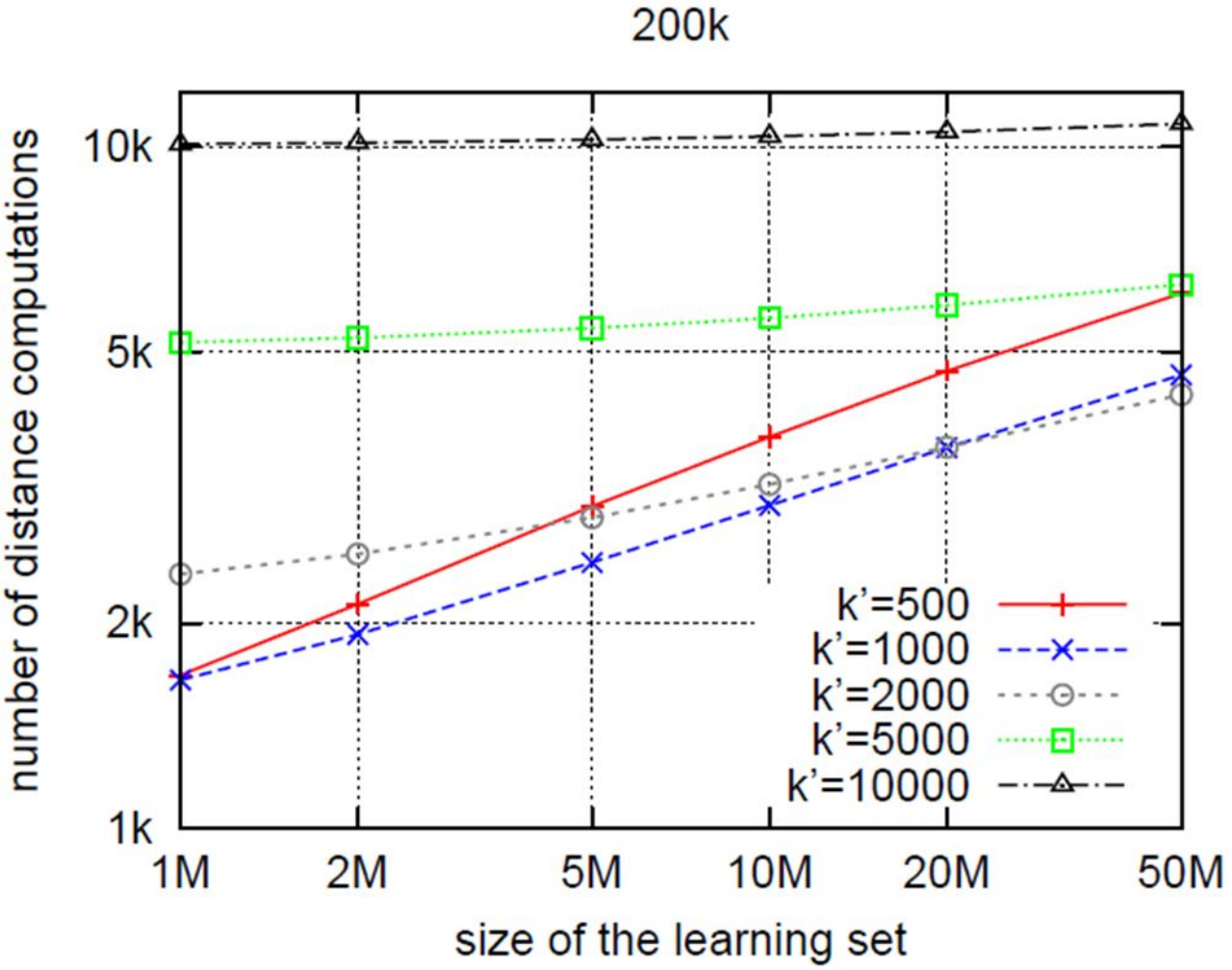
- Прямое сравнение дескриптора со всем словарём очень медленное
- Построим иерархическую структуру
  - Итеративно повторяем K-средних над словами и потом кластерами для построения дерева
  - Обучение дополнительных связей в графе на тестовой выборке

Анализ алгоритма



- Размер связей в графе в зависимости от размера обучающей выборки

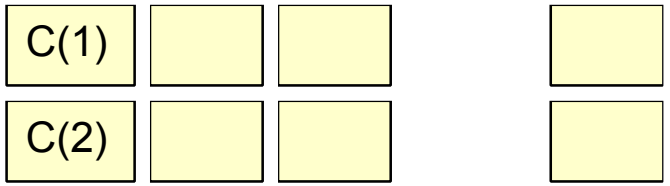
Анализ алгоритма



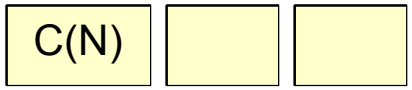
- Количество расчетов расстояний при поиске ближайших

Инвертированный индекс

- Вектор слов в дескрипторе очень разреженный
  - Например, 1k ненулевых элементов из 1M словаря



- Удобно хранить его в инвертированном индексе
  - Таблица (слова)x(изображения)
  - Список слов в словаре (терминов)
  - Для каждого слова храним список изображений, в котором слово встречается



- Ускорение поиска:
  - Самые частые слова идут в начале списка

Поиск по «мешку слов»

- Первый алгоритм CBIR над BoW готов!
  - Дескриптор «мешок слов» большой размерности (1M)
  - АКМ для построения словаря по большой коллекции (5к)
  - Инвертированный индекс для хранения

- Тестирование:
  - 5к+100к изображений, 1M слов, 1GB индекс, поиск в нём 0.1с
  - 5к+100к+1M изображений, 1M слов, 4GB+, хранение файла на диске, поиск 10-35с

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” CVPR, 2007.

Улучшение поиска

Microsoft Research

- Ранжирование списка результатов
- Раскрытие запросов
- Улучшение работы «мешка слов»
  - Коды Хэмминга
  - Слабая геометрия

Геометрическое сопоставление



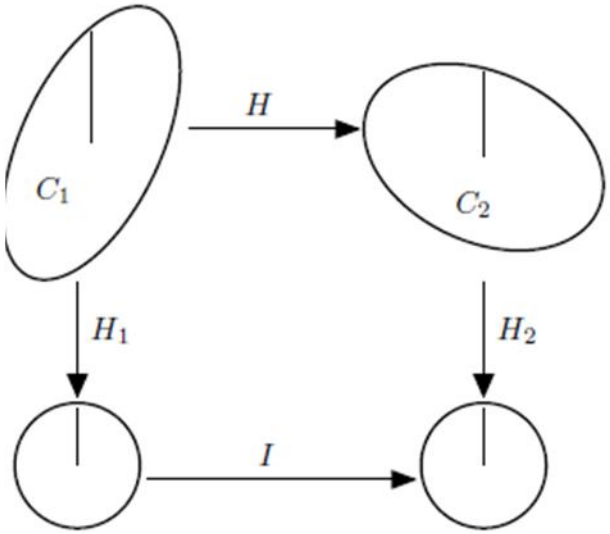

- Если мы решаем задачу «near-duplicate», то найденные изображения можно хорошо сопоставить с запросом
- Стандартная схема (локальные особенности + робастное вычисление преобразования) слишком медленное для полного перебора
- Можем использовать для постобработки – ранжирования найденных изображений

Microsoft Research

Схема ранжирования

- Сопоставим особенности между «запросом» и отфильтрованными поиском изображением
- Выбросы отфильтруем с помощью LO-RANSAC
  - Вначале простую модель
  - Затем по инлаерам более сложную модель
- Уточним хорошую модель по найденным инлаерам
  - Аффинная модель
- Отсортируем изображения
  - Для тех, которые сопоставились
    - В начало списка
    - Порядок по количеству инлаеров (чем больше – тем выше в списке)
  - Для тех, которые не сопоставились
    - В конец списка
    - Без изменения порядка

Оценка модели по 1 паре



- $H = H_2^{-1}H_1$

- Достаточно одной пары соответствующих точек для генерации гипотезы
- Можно оценить до 5 параметров
  - Сдвиг (2)
  - Масштаб (1)
  - Поворот (1)
  - Пропорции (эллипсоид)

Модели преобразования

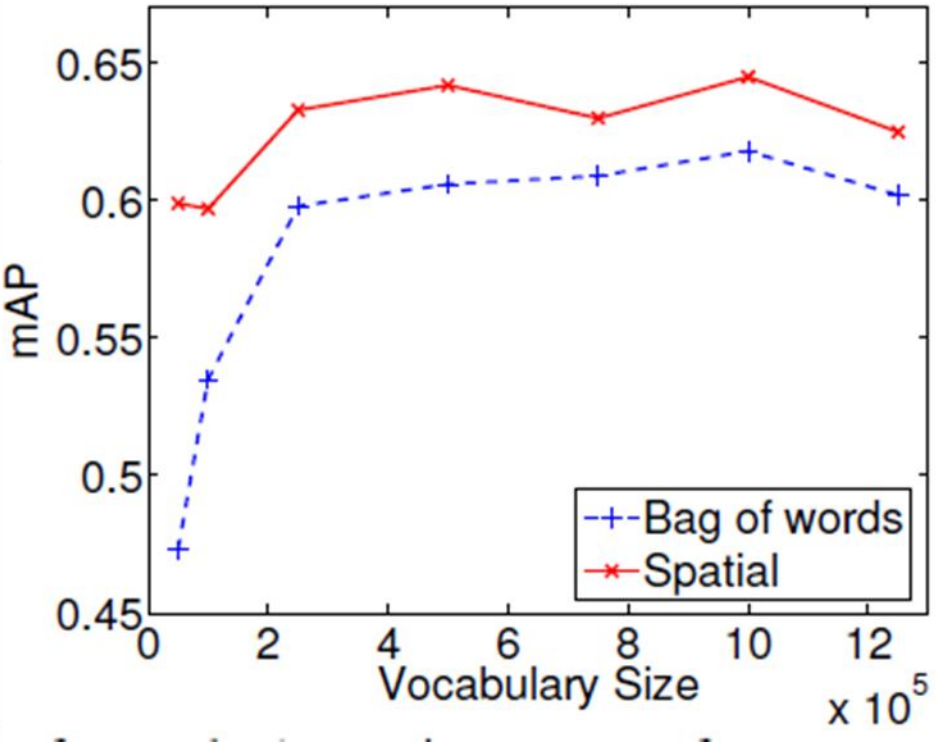
Microsoft Research

Transformation	dof	Matrix
translation + isotropic scale	3	$\begin{bmatrix} a & 0 & t_x \\ 0 & a & t_y \end{bmatrix}$
translation + anisotropic scale	4	$\begin{bmatrix} a & 0 & t_x \\ 0 & b & t_y \end{bmatrix}$
translation + vertical shear	5	$\begin{bmatrix} a & 0 & t_x \\ b & c & t_y \end{bmatrix}$

- 3dof
  - Моделируем масштабирование / расстояние до объекта
- 4dof
  - Моделируем ракурсы
- 5dof
  - Сохраняем «вертикальность», моделируем «скос» (shear)

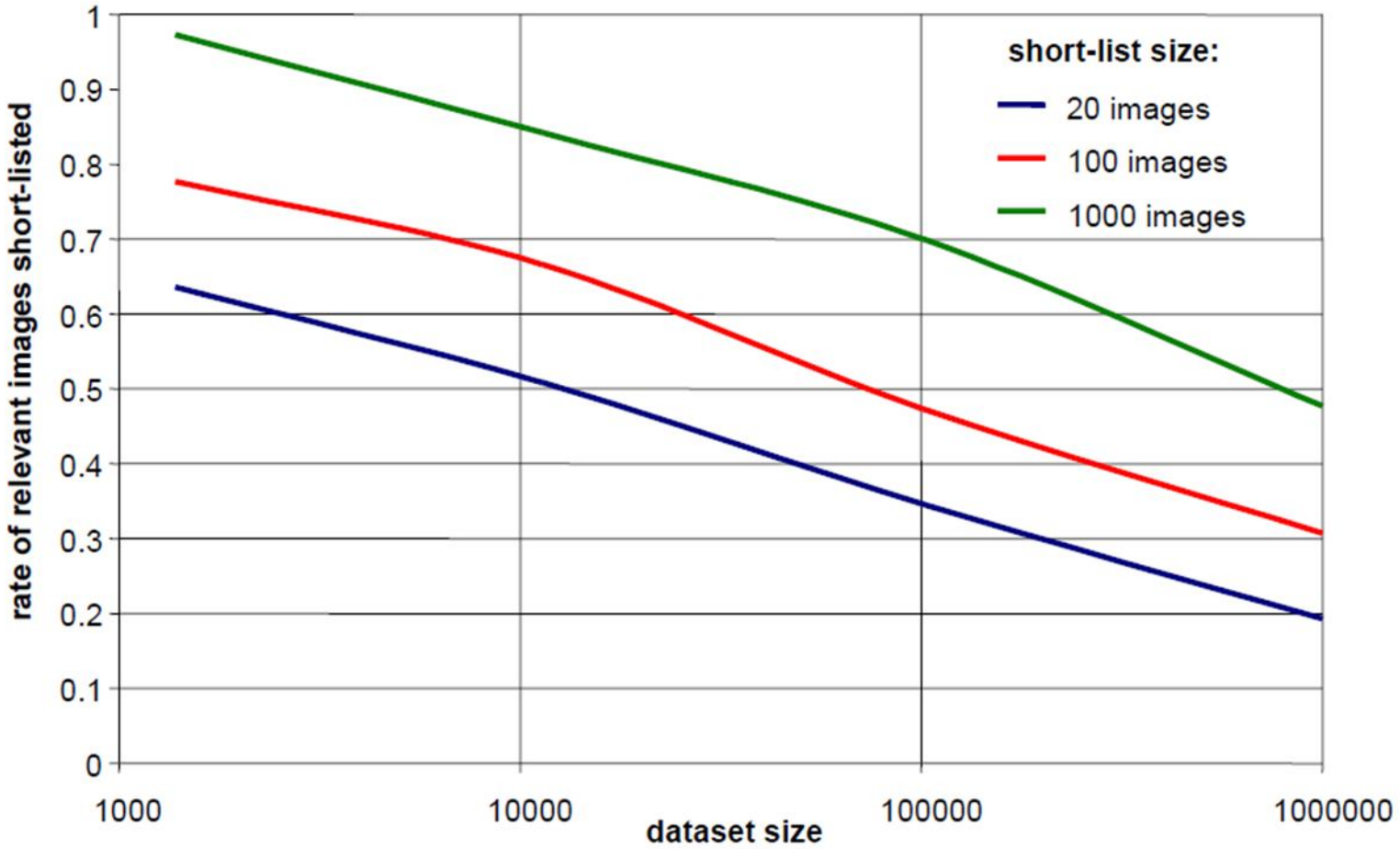
Результаты ранжирования

Vocab Size	Bag of words	Spatial
50K	0.473	0.599
100K	0.535	0.597
250K	0.598	0.633
500K	0.606	0.642
750K	0.609	0.630
1M	**0.618**	**0.645**
1.25M	0.602	0.625



- Для поиска изображений архитектуры ранжирование показывает существенный прирост в точности

Результаты ранжирования



- Доля нужных изображений в верхней части ранжированного списка после геометрического сопоставления

Раскрытие запросов

Microsoft Research

- Transitive closure expansion (TCE)
  - Строим дерево запросов
  - Вершина – исходный запрос
  - Потомки – наиболее хорошо сопоставленные изображения из ответа на запрос

- Additive query expansion (AQE)
  - Отображаем интересные точки с найденных изображений на исходное
  - Используем модифицированное изображение для поиска и дополнения результатов

Результаты

Microsoft Research

re-ranking method:	Oxford+ 0	Oxford+ 10k	Oxford+ 100k	Oxford+ 1M	Holidays
geometric verification	0.667	0.652	0.591	0.486	0.848
TCE	0.757	0.735	0.674	0.582	0.827
AQE	0.747	0.736	0.687	0.572	0.842

Microsoft Research

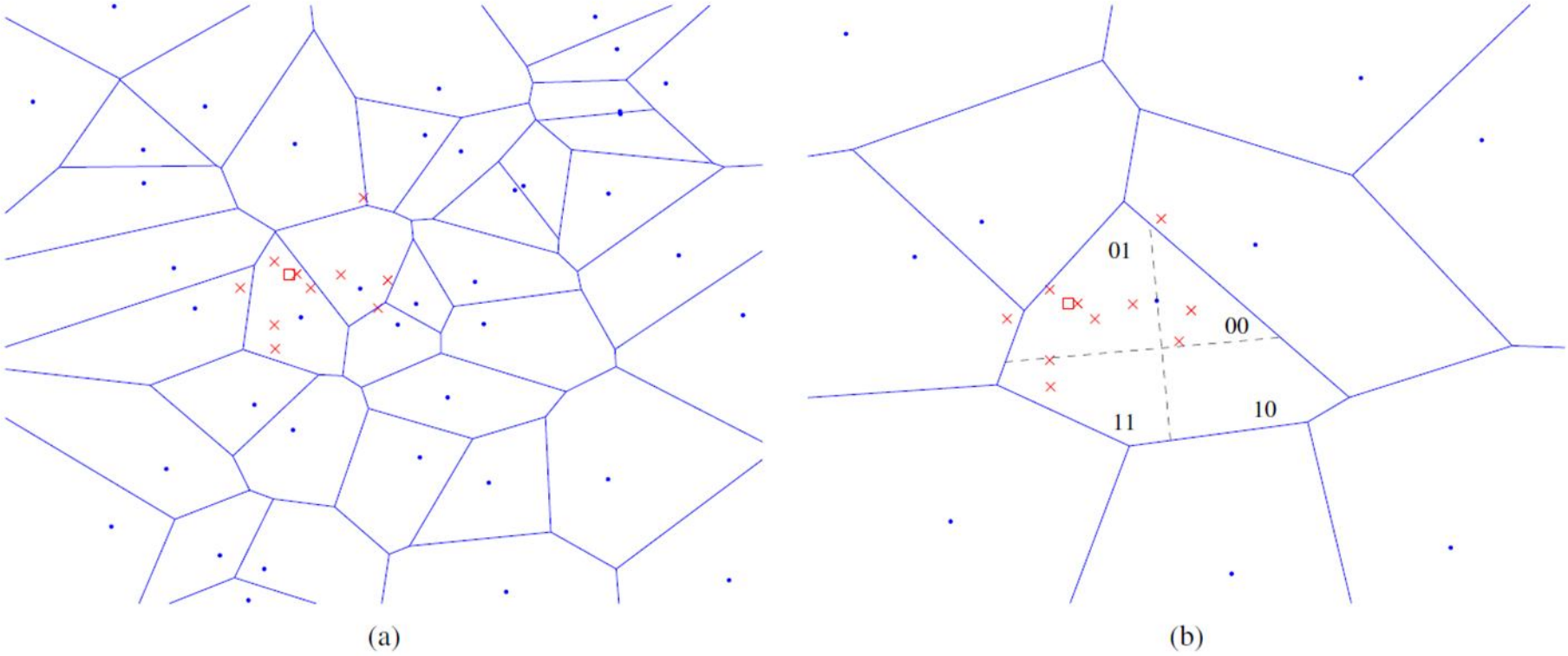
Мешок слов и сопоставление

- Сопоставление изображений
  - Находим локальные особенности, считаем дескриптор
  - Находим ближайшие пары по дескрипторам
  - Считаем количество пар, с ошибкой меньше эпсилон

- Сопоставление по «мешку слов»
  - Находим локальные особенности, считаем дескриптор
  - Квантуем локальные особенности по словарю
  - Считаем для каждого изображения гистограмму частот
  - Сравниваем гистограммы частот

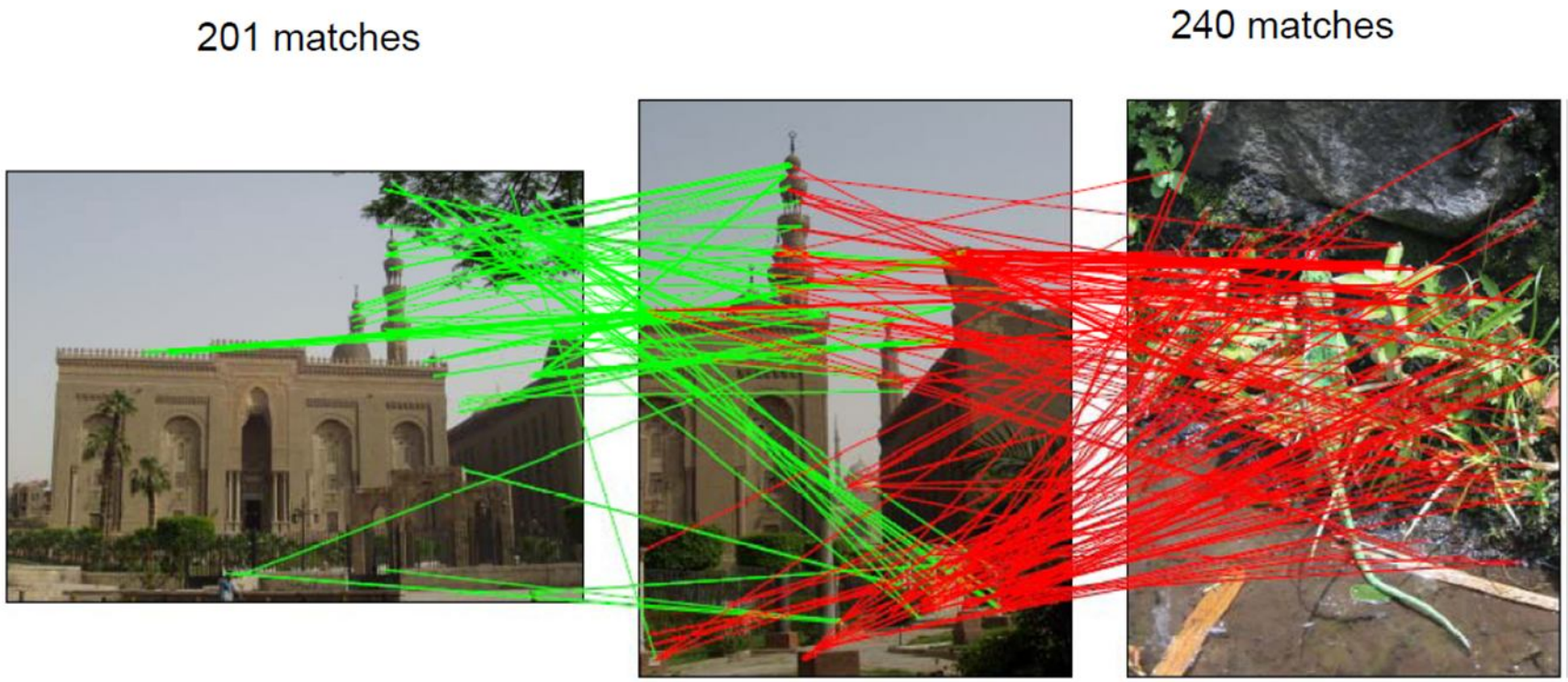
- Практически эквивалентно!
  - В первом случае сравниваем дескрипторы непосредственно друг с другом
  - Во втором случае сравниваем точки по номерам в словаре

Диаграмма Вороного



- Маленький словарь – большие ячейки
  - Слишком грубый порог на сравнение!
- Большой словарь – маленькие ячейки
  - Слишком точный порог на сравнение!

Пример

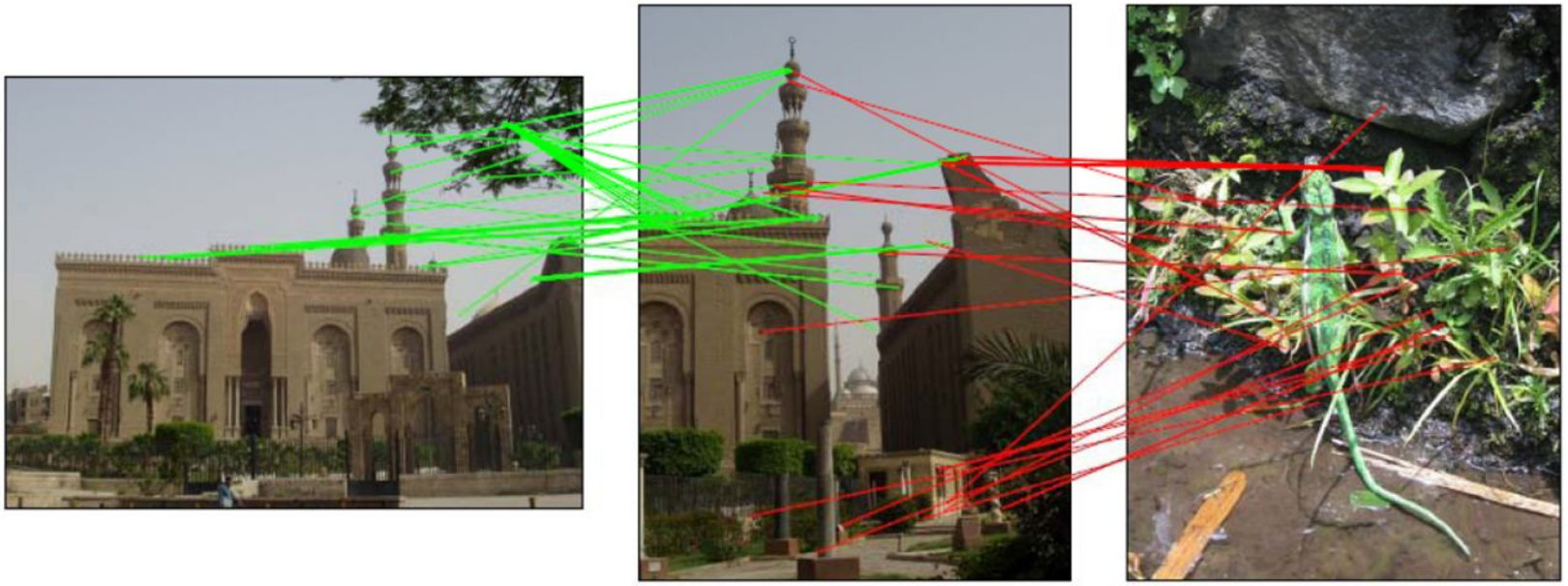


- 20K словарь

Пример

69 matches

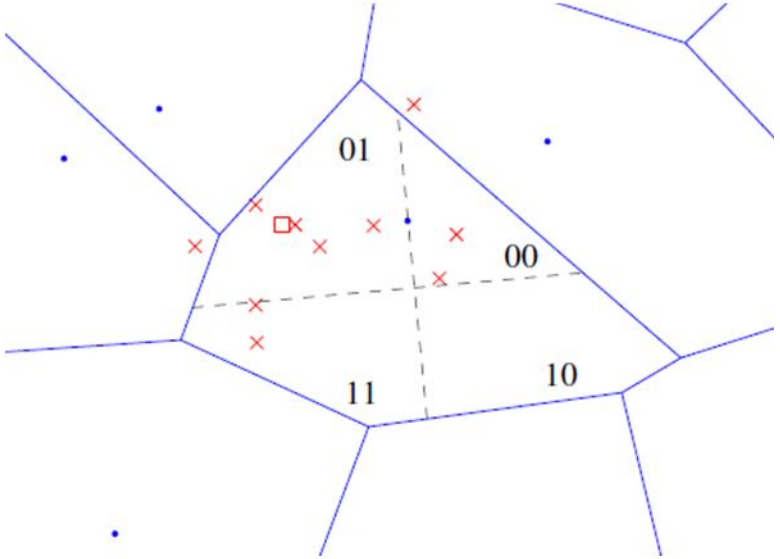
35 matches



- 200K словарь

Hamming Embedding

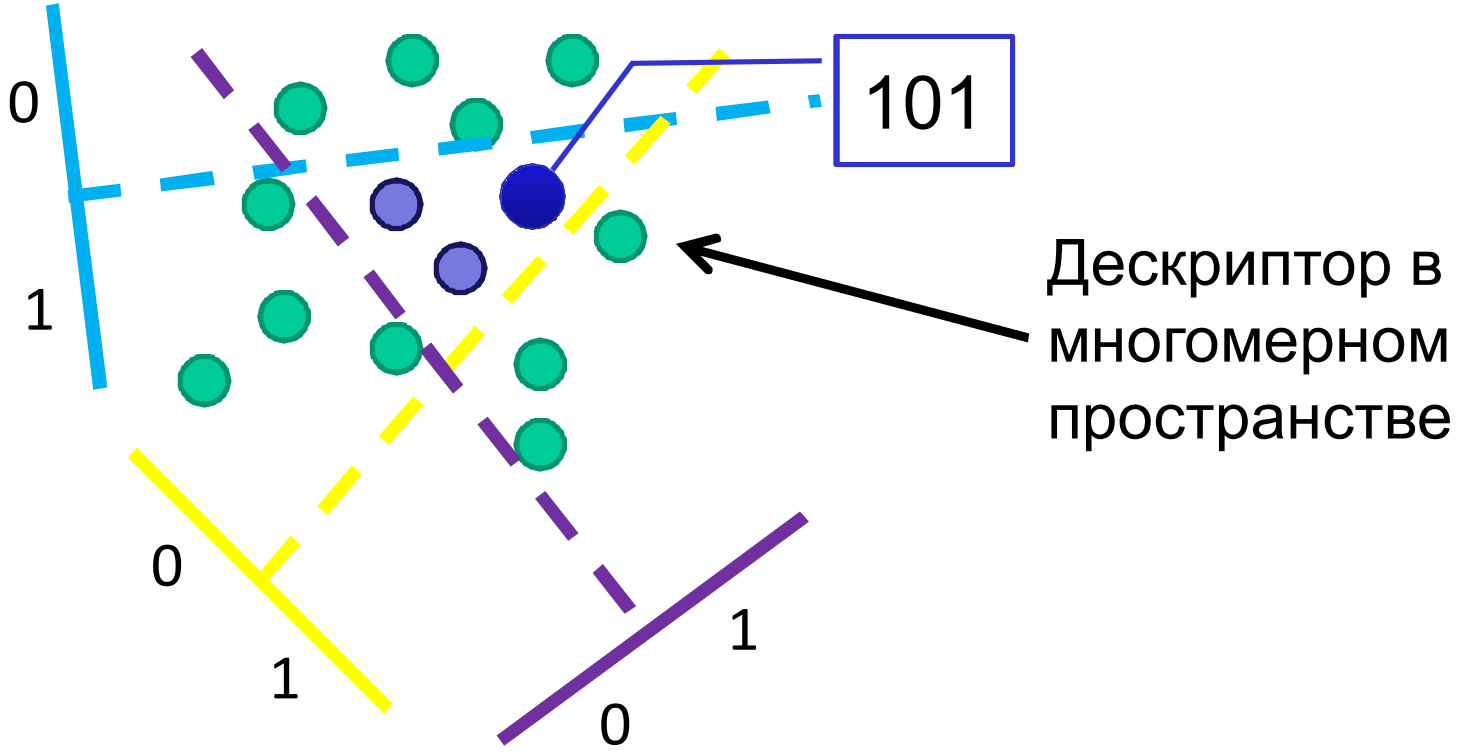
- Хотим записать не только #слова для особенности из изображения, но и описать положение внутри ячейки (доп.код)
- Будем сравнивать тогда не только по номеру, но и по доп.коду
- Код должен быть маленьким, и сравнение быстрое!
  - Построим бинарный код
  - Сравнивать будем по расстоянию Хэмминга



Locality Sensitive Hashing (LSH)

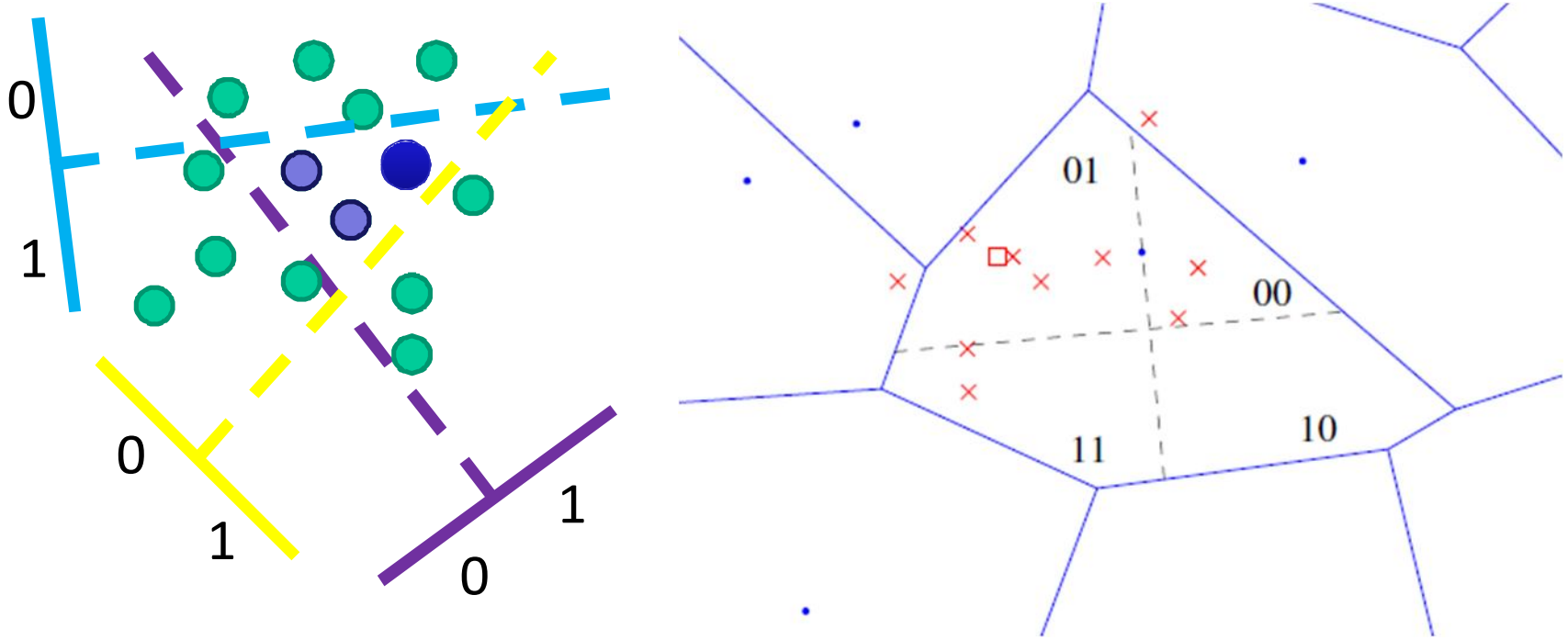
Microsoft Research

- Возьмем случайную проекцию данных на прямую
- Случайно выберем порог, пометив проекции 0 или 1 (1 бит подписи)
- С увеличением числа бит код приближает L2-метрику в исходных дескрипторах



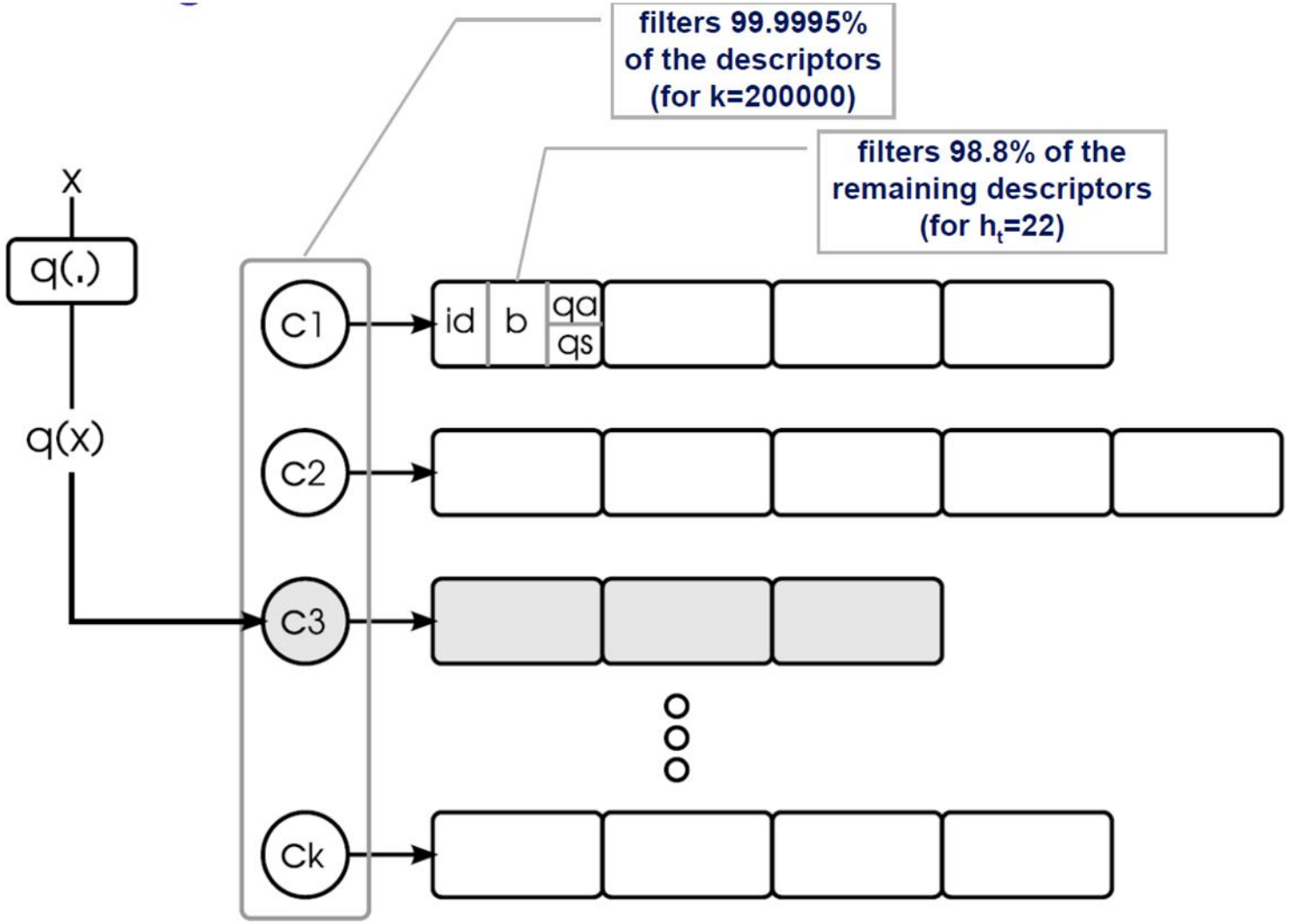
A. Andoni and P. Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In FOCS, 2006

Hamming Embedding



- Возьмем все дескрипторы, попавшие в одну ячейку
- Сгенерируем n случайных прямых (направлений проецирования)
- Спроецируем все дескрипторы на прямую
- Выберем точку на прямой (порог) таким образом, чтобы справа и слева было поровну точек
  - Такой код будет оптимальным

Модификация индекса

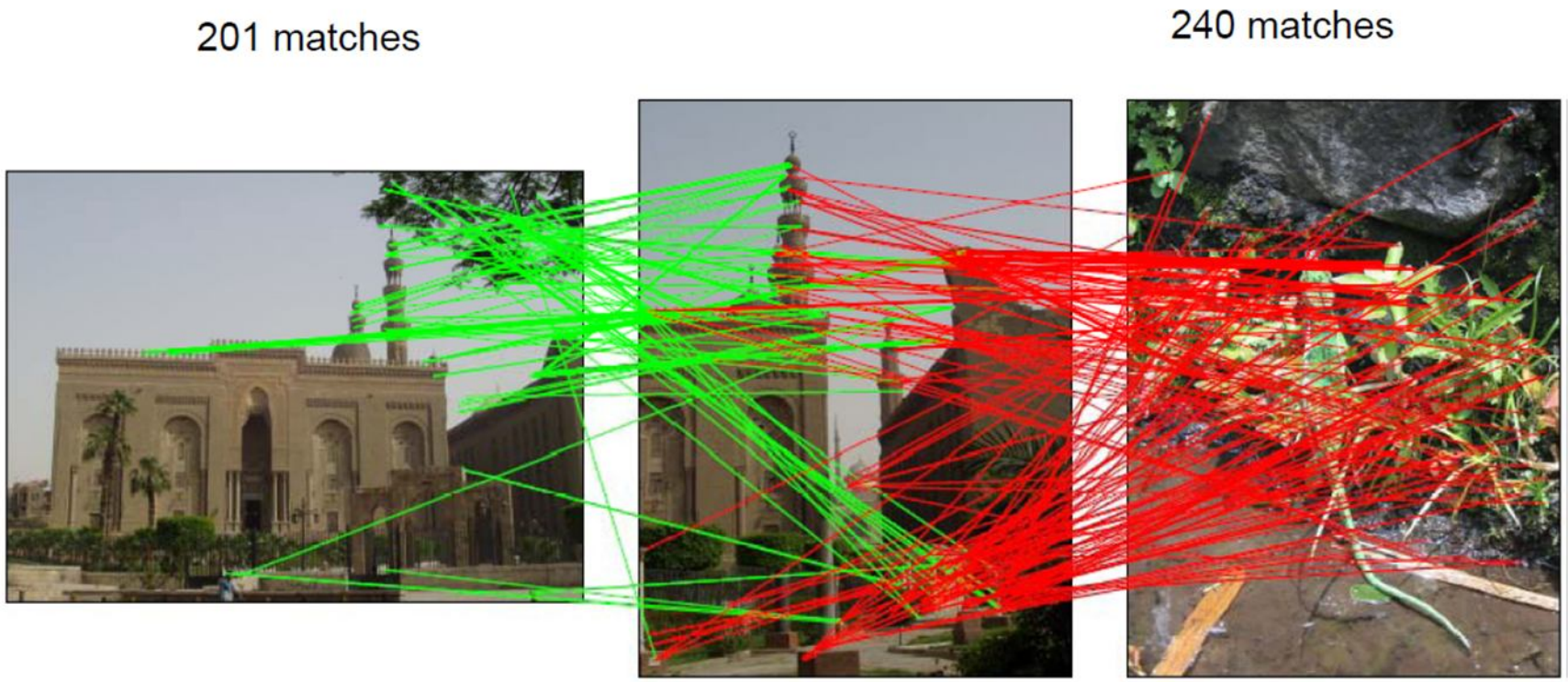


Microsoft Research

Алгоритм

- Для каждого дескриптора:
  - Квантуем по словарю (#слова)
  - Вычисление бинарного кода
- Считаем точки сопоставленными, только если выполняются оба условия:
  - #слов совпадают
  - Коды по расстоянию Хэмминга отличаются не более чем на z

Пример

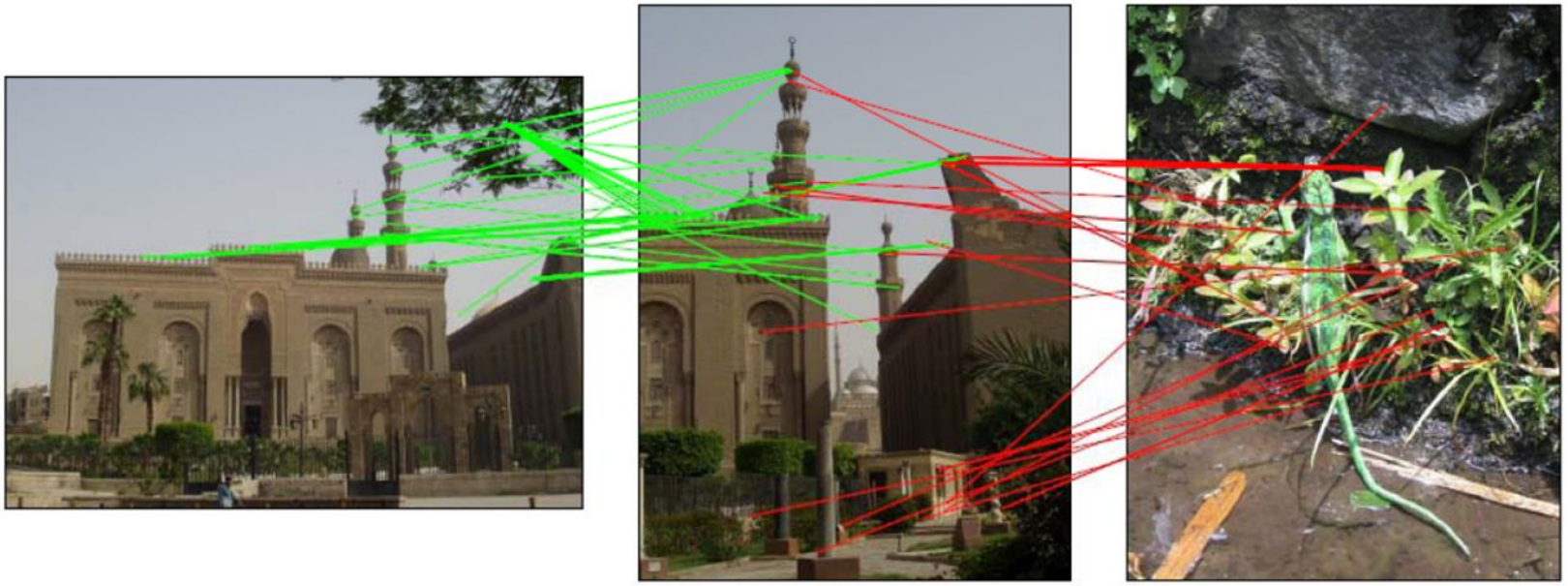


- 20K словарь

Пример

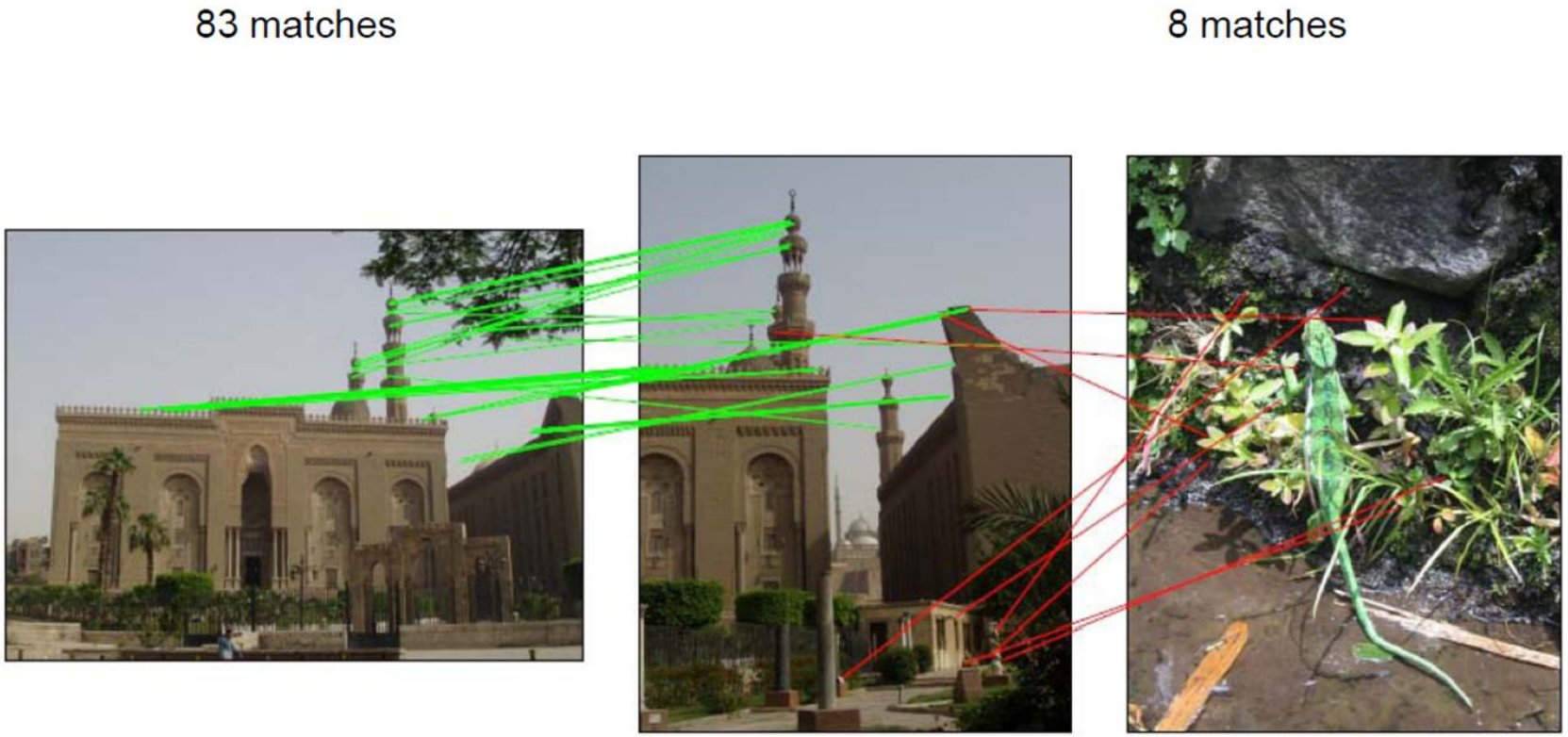
69 matches

35 matches



- 200K словарь

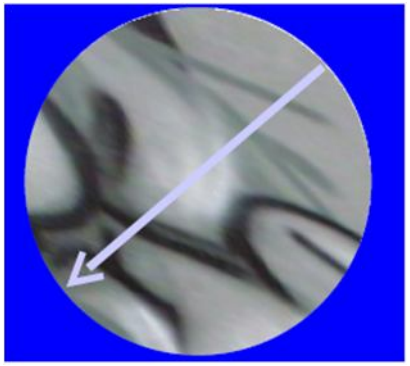
Hamming Embedding

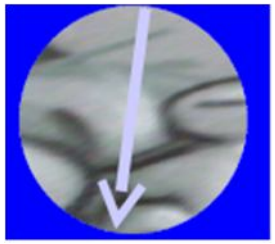


- Соответствий с правильным изображением в 10 раз больше, чем с неправильным!

«Слабая геометрия»

- Каждая характерная точка определяется в т.ч. масштабом (характерным размером) и ориентацией
- Пример:





- 20 градусов разницы по ориентации
- Масштаб в 1.5 раза

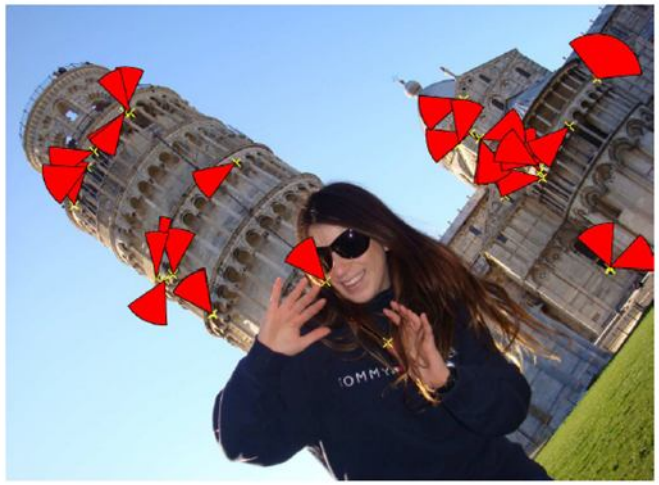
- Каждая сопоставление задаёт разницу по углу и масштабу
- Для изображения в целом изменение должны быть согласованны

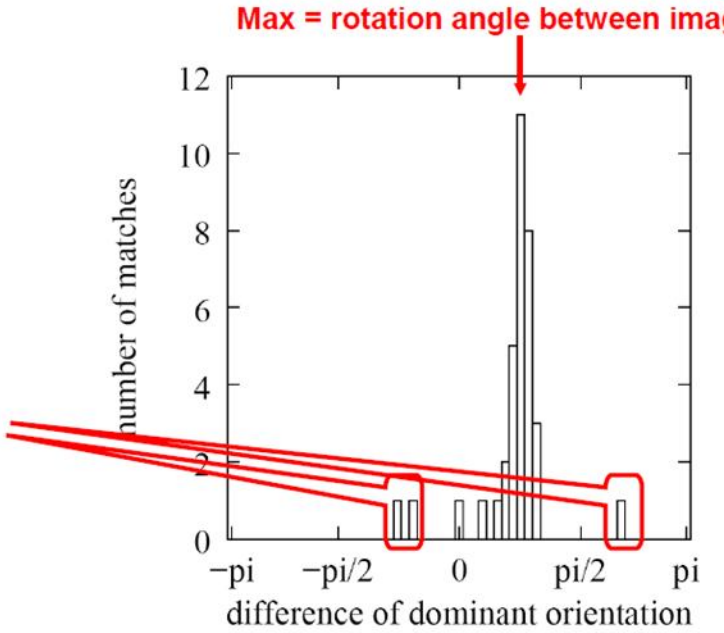
Примеры

Pisa tower: Let analyze the dominent orientation difference of matching descriptors

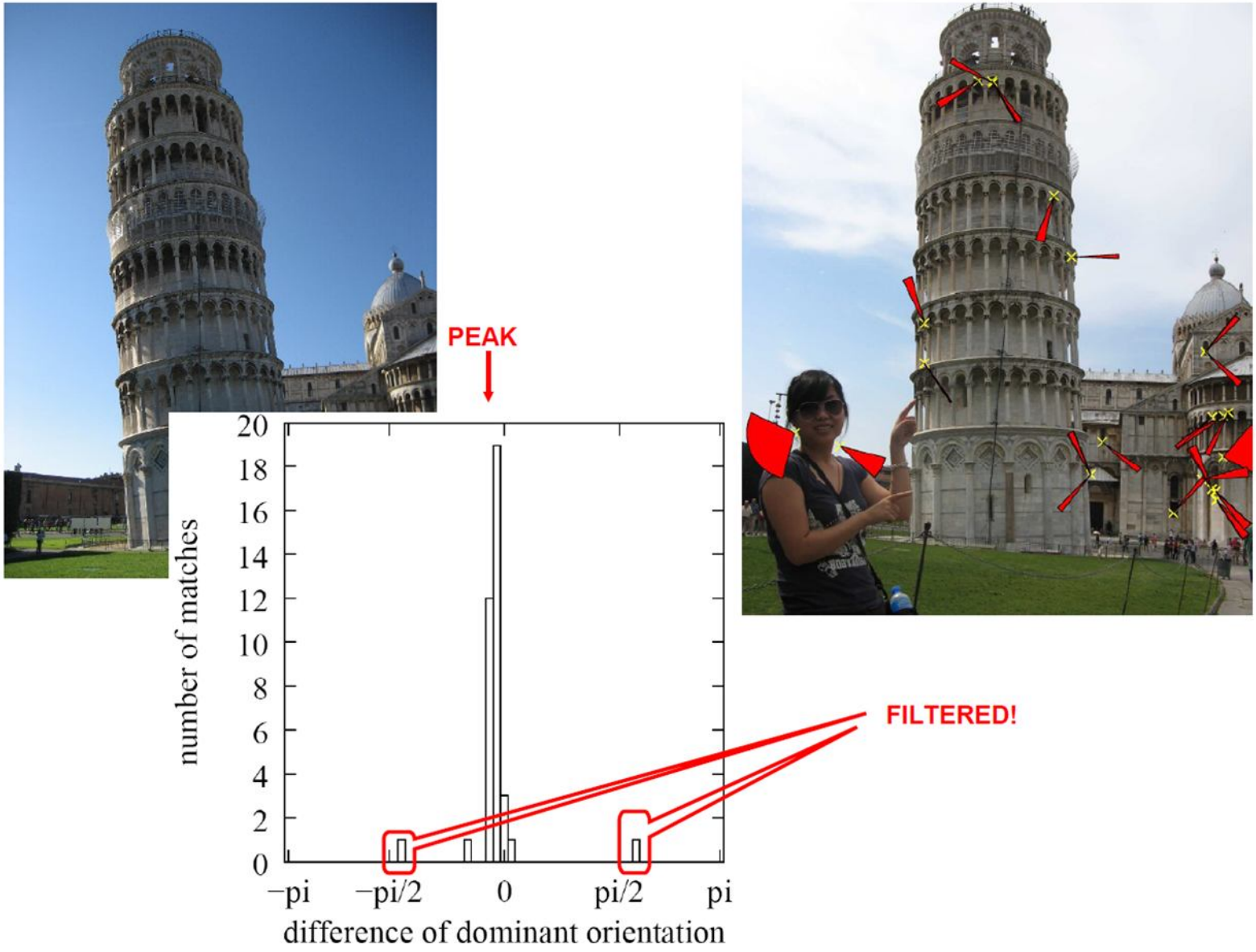
Примеры

Orientation consistency

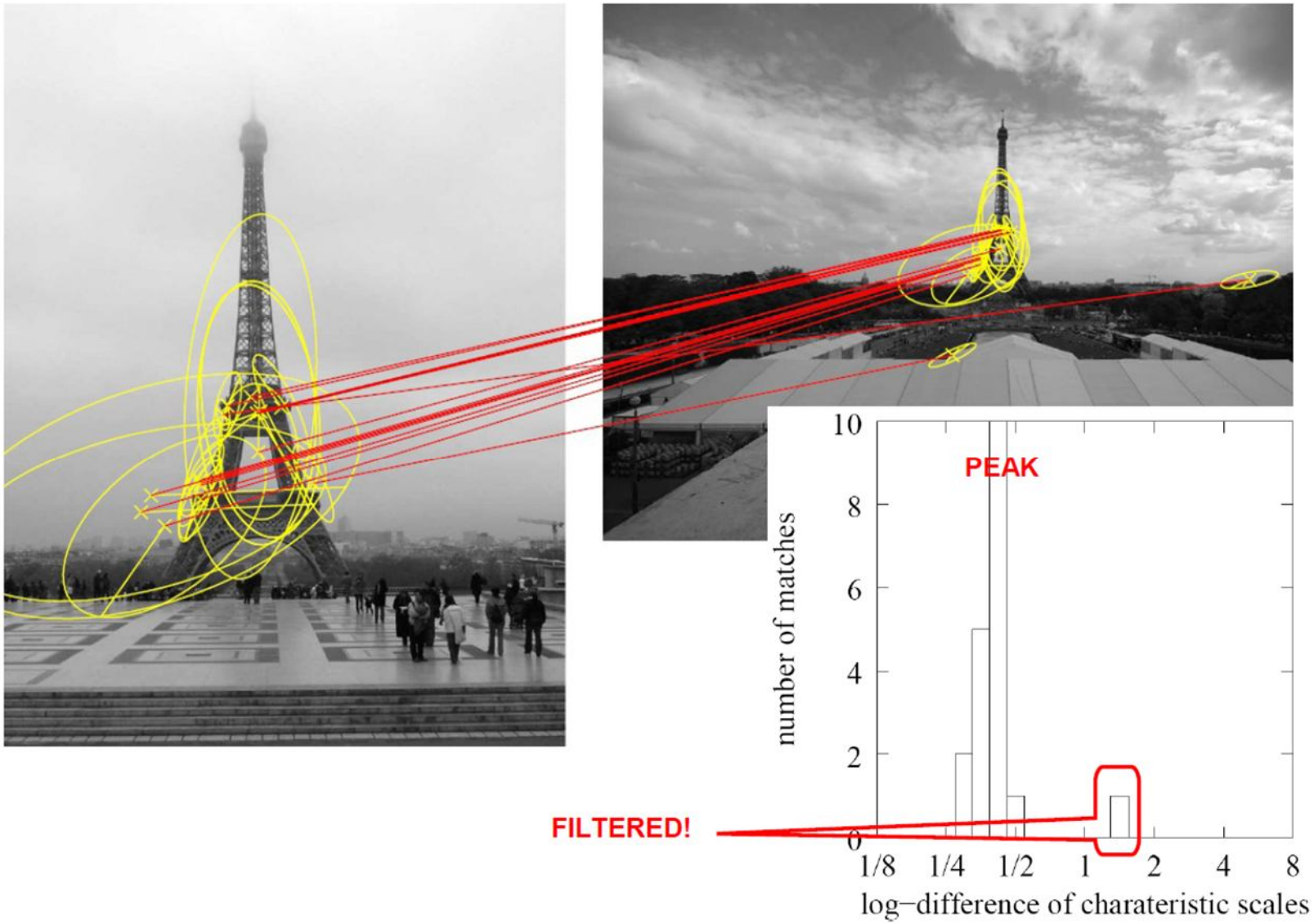




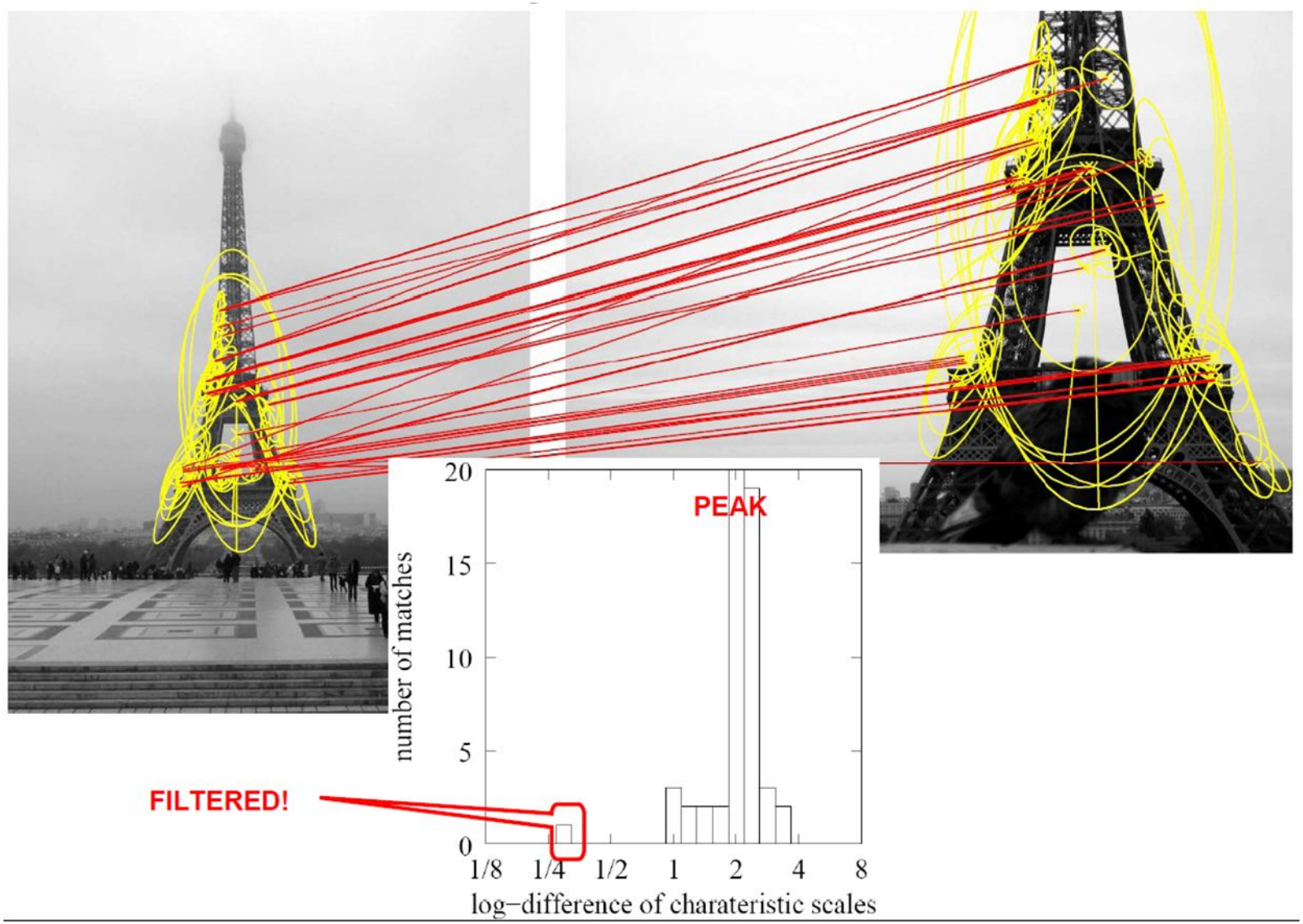
Примеры


PEAK
20
18
16
14
12
10
8
6
4
2
0
number of matches
−pi
−pi/2
0
pi/2
pi
difference of dominant orientation
FILTERED!

Примеры


PEAK
FILTERED!
number of matches
10
8
6
4
2
0
1/8
1/4
1/2
1
2
4
8
log−difference of charateristic scales

Примеры

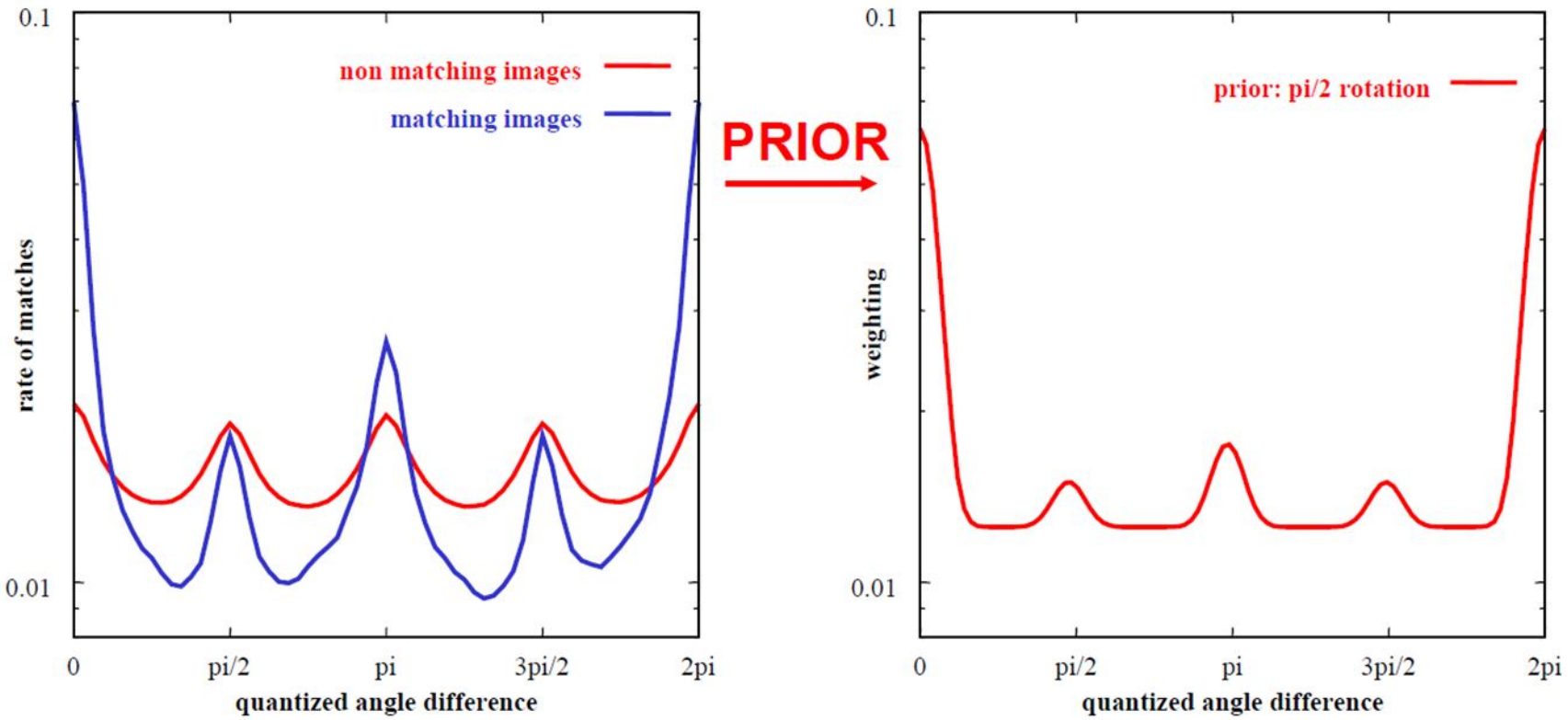

PEAK
FILTERED!
number of matches
20
15
10
5
0
1/8
1/4
1/2
1
2
4
8
log−difference of charateristic scales

Microsoft Research

Смысл

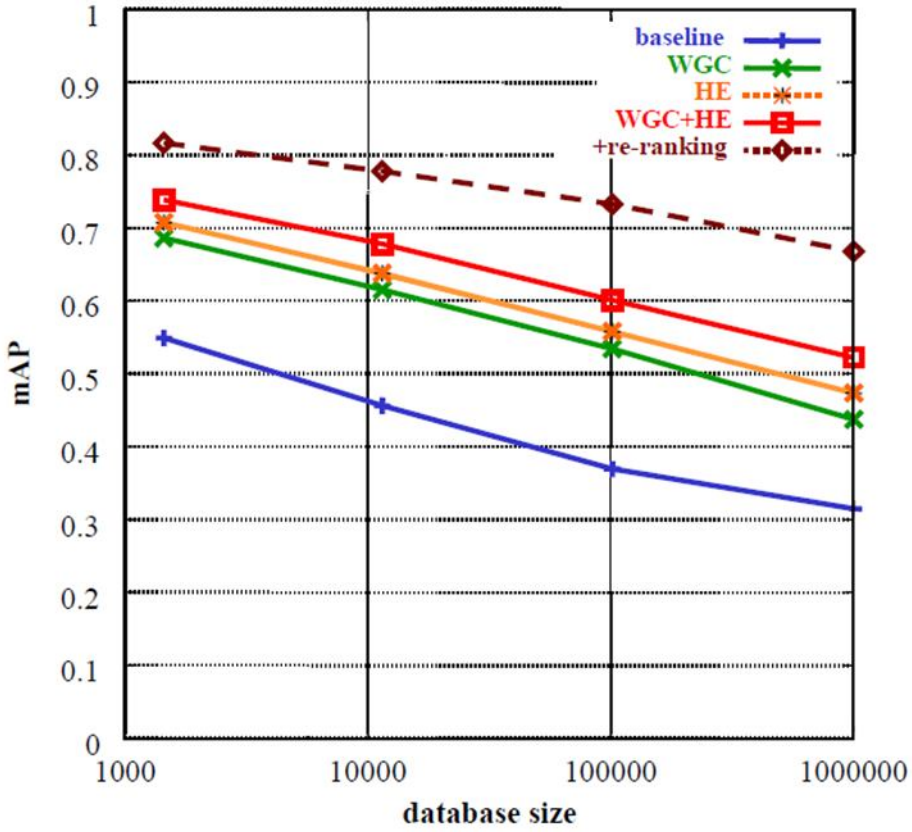
- Масштаб и ориентация «примерно» не зависят друг от друга
- Голосование с учётом дискретного масштаба и поворота
  - Отдельный вес для каждой комбинации (угол/поворот)
    - Фактически, гистограмма
  - Берем максимумы по углу / масштабу
  - Берём из них минимум
- Только соответствия, согласованные по изменению масштаба и ориентации вносят вклад в финальную оценку

Априорные данные



- Люди фотографируют одну и ту же сцену с одинаковых ракурсов

Суммарный результат



Average query time (4 CPU cores)	
Compute descriptors	880 ms
Quantization	600 ms
Search – baseline	**620** ms
Search – WGC	**2110** ms
Search – HE	**200** ms
Search – HE+WGC	**650** ms

- Каждый элемент – weakly geometry, hamming embedding ранжирование по геометрии существенно повышает точность
- При этом совместно использование WGC и HE позволяет достичь скорости, сравнимой с базовым методом!

Резюме «мешка слов»

Microsoft Research

- Большой словарь
  - + Слабая геометрия
  - + Hamming embedding
- Инвертированный индекс
- Ранжирование по сопоставлению
- Раскрытие запросов

Microsoft Research

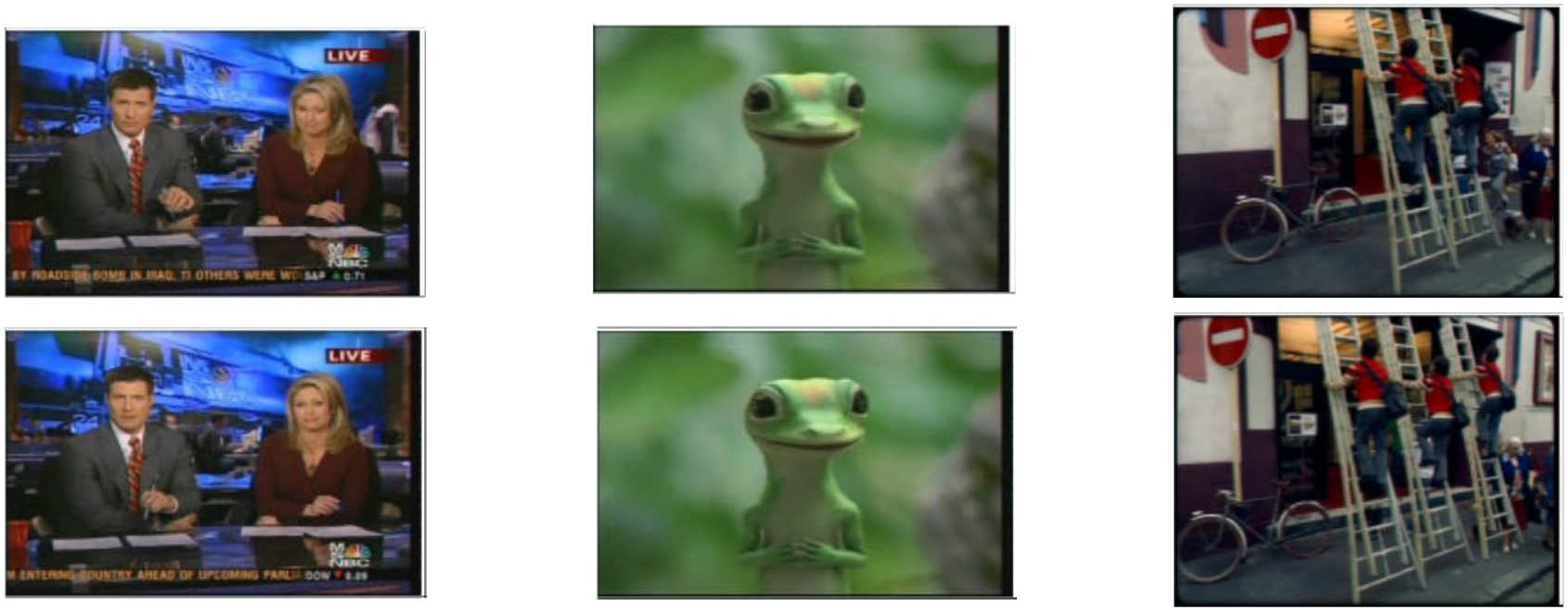
Построение подписи

- Отойдём от инвертированного индекса

- Будем строить короткие «подписи» для каждого изображения с помощью «хэш-функций»
  - min-Hash
  - Семантическое хэширование
  - Локально-чувствительное хэширование
  - Обучение подписи
  - Спектральное хэширование
  - Обучение метрик и подписей

min-Hash

- Почти одинаковые изображения (шум, движение, и т.д.)
- Большие базы данных
- Быстрый поиск (линейный по числу дубликатов)



O.Chum, J.Philbin, A.Zisserman. Near Duplicate Image Detection: min-Hash and tf-idf Weighting, BMVC 2008

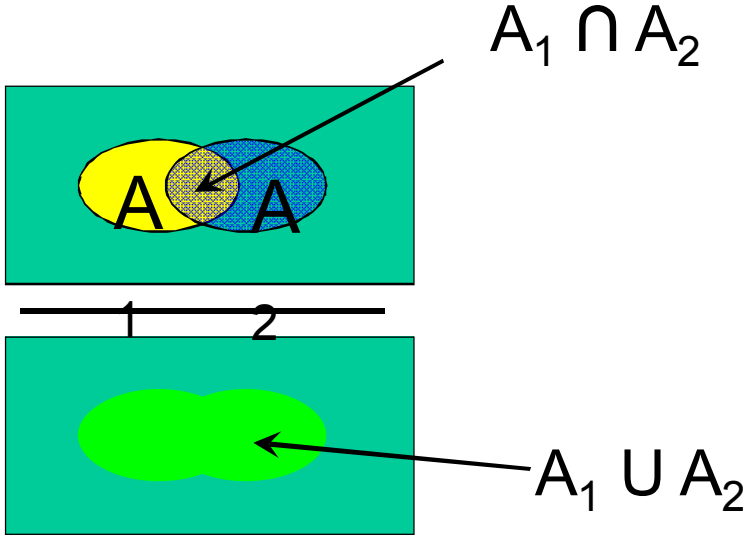
min-Hash

- Min-Hash - локально-чувствительная хэш-функция *m* (locality sensitive hashing,LSH), которая выбирает элементы $m(A_1)$ из набора A_1 и $m(A_2)$ из набора A_2 , таким образом, что выполняется:

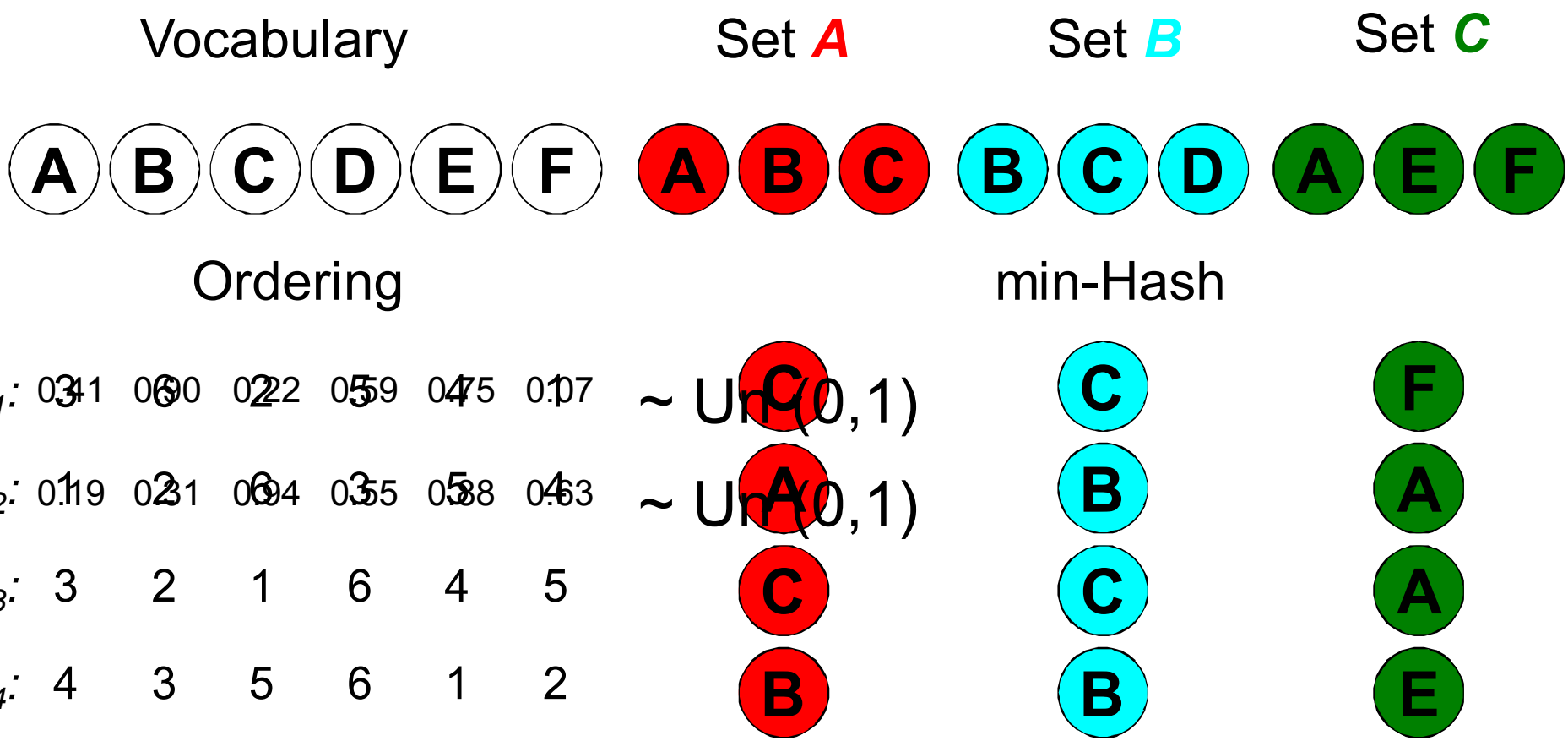
$$P\{m(A_1) == m(A_2)\} = \text{sim}(A_1, A_2)$$

- Похожие изображения должны иметь пересекающиеся наборы визуальных слов
- Сходство изображений измеряем как пересечение множеств с помощью min-Hash алгоритма

$$\text{sim}(\mathcal{A}_1, \mathcal{A}_2) = \frac{|\mathcal{A}_1 \cap \mathcal{A}_2|}{|\mathcal{A}_1 \cup \mathcal{A}_2|} \in \langle 0, 1 \rangle$$

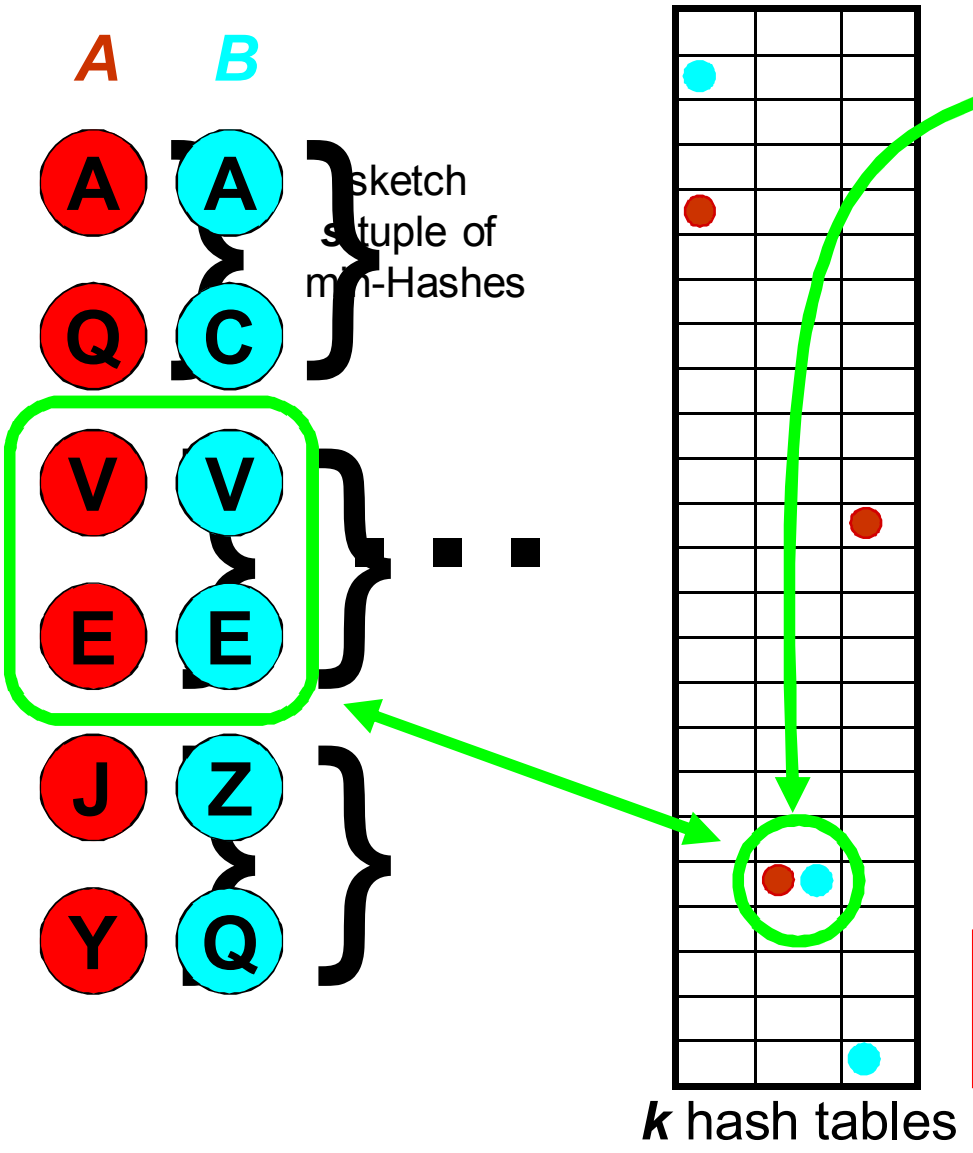


min-Hash



overlap (*A*,*B*) = 3/4 (1/2) overlap (*A*,*C*) = 1/4 (1/5) overlap (*B*,*C*) = 0 (0)

min-Hash Retrieval



Sketch collision

s – размер sketch

k – количество хэш-таблиц

Вероятность коллизии:

$$sim(A, B)^s$$

Вероятность извлечь
(минимум 1 коллизия скетча)

$$1 - (1 - sim(A, B)^s)^k$$

Microsoft Research

Более сложные метрики

• Модель «**Set of words»** с весами
• Вес d_w задаёт «важность» слова X_w

$$\mathrm{sim}_w(\mathcal{A}_1, \mathcal{A}_2) = \frac{\sum_{X_w \in \mathcal{A}_1 \cap \mathcal{A}_2} d_w}{\sum_{X_w \in \mathcal{A}_1 \cup \mathcal{A}_2} d_w}$$

• Модель «**Bag of words»**
• Храним и частот слов
• Метрика - пересечение гистограмм

$$\mathrm{sim}_{h_0}(\mathcal{A}_1, \mathcal{A}_2) = \frac{\sum_w \min(t_1^w, t_2^w)}{\sum_w \max(t_1^w, t_2^w)}$$

• Модель **Bag of words,** но с учетом весов для слов:

$$\mathrm{sim}_h(\mathcal{A}_1, \mathcal{A}_2) = \frac{\sum_w d_w \min(t_1^w, t_2^w)}{\sum_w d_w \max(t_1^w, t_2^w)}$$

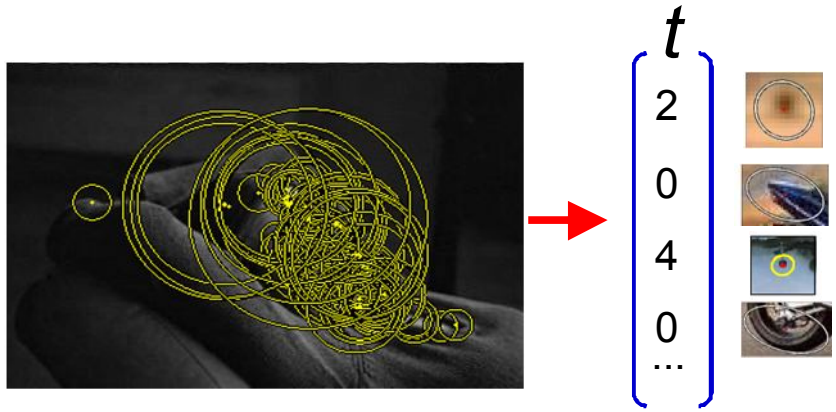
TF-IDF

Term Frequency – Inverse Document Frequency (*tf-idf*) weighting scheme

$$idf_W = \log \frac{\#\ \text{documents}}{\#\ \text{docs containing } X_W}$$

Слова, встречающиеся во многих документах, малоинформативны

Записываем частоты слов (хорошо для повторяющихся структур и т.д.)



[1] Baeza-Yates, Ribeiro-Neto. Modern Information Retrieval. ACM Press, 1999.
[2] Sivic, Zisserman. Video Google: A text retrieval approach to object matching in videos. ICCV'03.
[3] Nister, Stewenius. Scalable recognition with a vocabulary tree. CVPR'06.
[4] Philbin, Chum, Isard, Sivic, Zisserman. Object retrieval with large vocabularies and fast spatial matching. CVPR'07.

Добавление весов в min-Hash

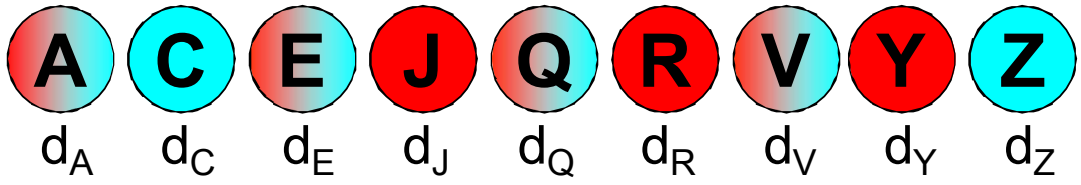
Для исходной min-hash функции $f_j(X_w) = x \qquad x \sim \mathrm{Un}(1,0)$

У всех слов X_w одинаковая вероятность быть min-Hash

Для функции $f_j(X_w) = \dfrac{-\log x}{d_w} \qquad x \sim \mathrm{Un}(1,0)$

Вероятность для X_w быть min-Hash пропорциональна d_w

A U ***B***:



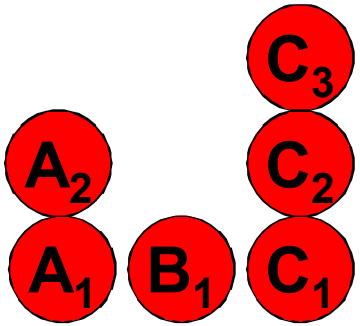
$$P(m(\mathcal{A}) = m(\mathcal{B})) = \frac{\sum_{X_w \in \mathcal{A} \cap \mathcal{B}} d_w}{\sum_{X_w \in \mathcal{A} \cup \mathcal{B}} d_w}$$

Пересечение гистограмм с min-Hash

Идея: представим гистограмму как набор, затем используем min-Hash для наборов

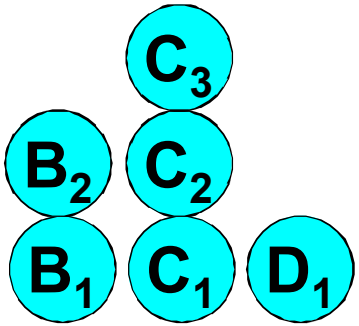
Слова: A B C D

Мешок слов ***A*** / набор ***A'***



$t_A = (2,1,3,0)$

Мешок слов ***B*** / набор ***B'***



$t_B = (0,2,3,1)$

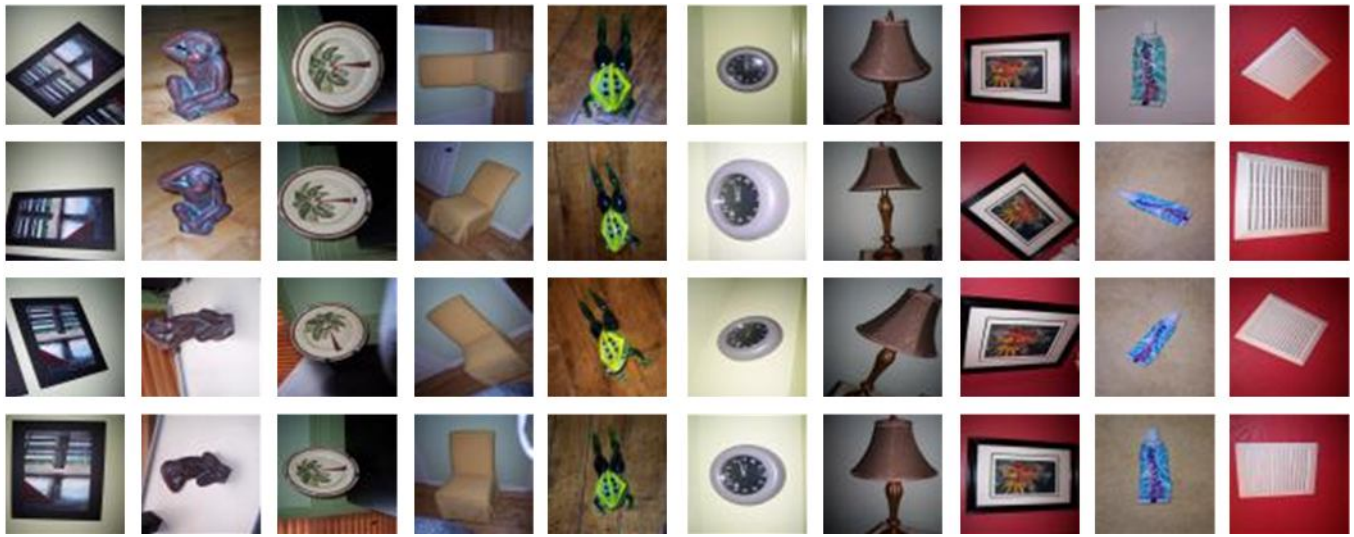
min-Hash словарь: A_1 A_2 B_1 B_2 C_1 C_2 C_3 D_1

A' U ***B'***: A_1 A_2 B_1 B_2 C_1 C_2 C_3 D_1

Пересечение множеств ***A'*** и ***B'*** = пересечение гистограмм ***A u B***

Тестирование метода

- Данные - University of Kentucky Dataset
- 10,200 изображений по 4 в группе
- Запрашиваем последовательно изображение из каждой группы
- Измеряем количество правильно найденных из 4х самых верхних в списке найденных



Примеры запросов

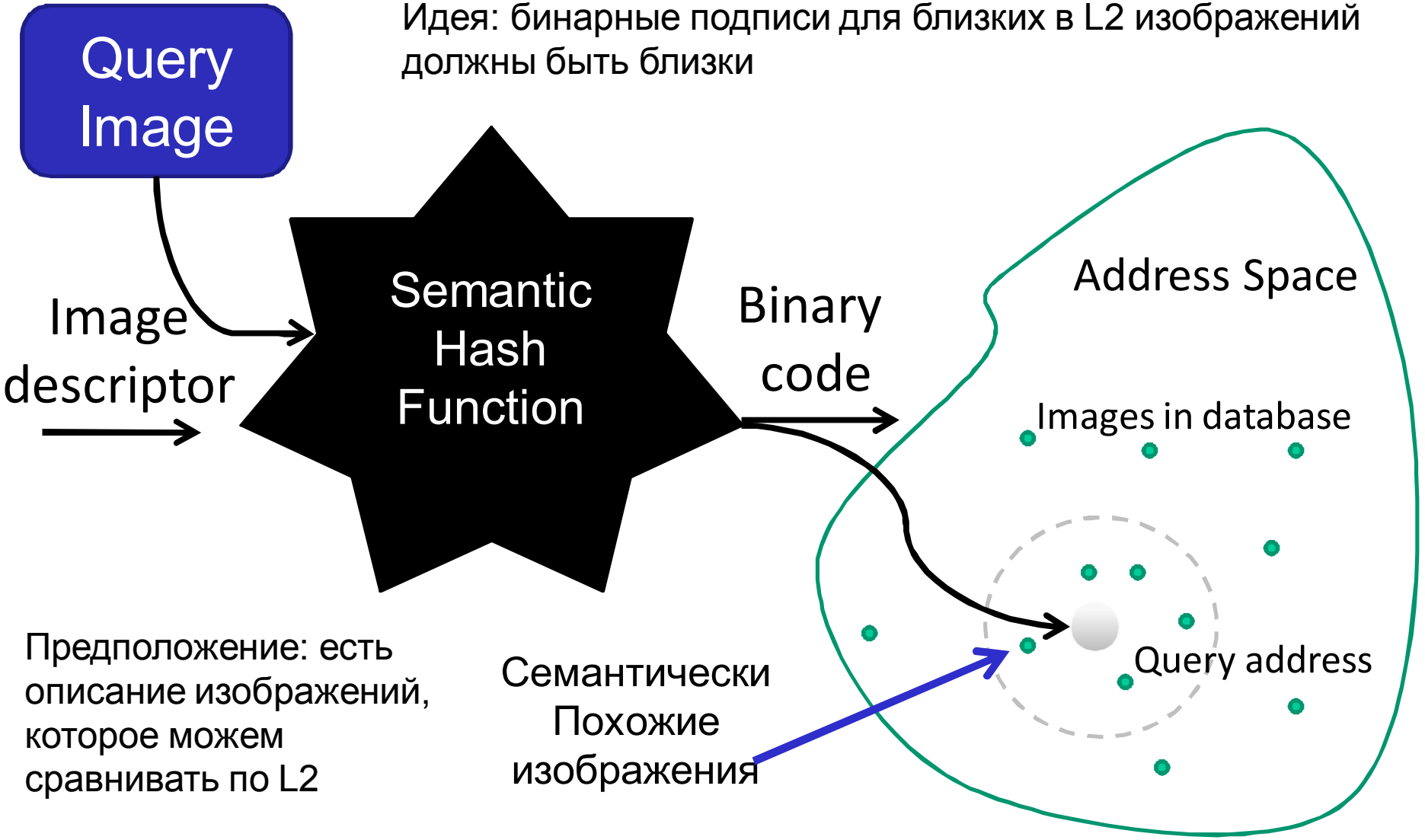
Query image:

- Наибольшую точность показывает использование взвешенной гистограммы
- Также при этом меньше возникает коллизий в хэшах

Результаты:
Set overlap, weighted set overlap, weighted histogram intersection

Семантическое хеширование

Идея: бинарные подписи для близких в L2 изображений должны быть близки



R. R. Salakhutdinov and G. E. Hinton. Semantic hashing. In SIGIR workshop on Information Retrieval and applications of Graphical Models, 2007.

Формализация

- Смысл:
  - Имеем x,y – вектора (дескрипторы)
  - Хотим получить бинарный код h(x)

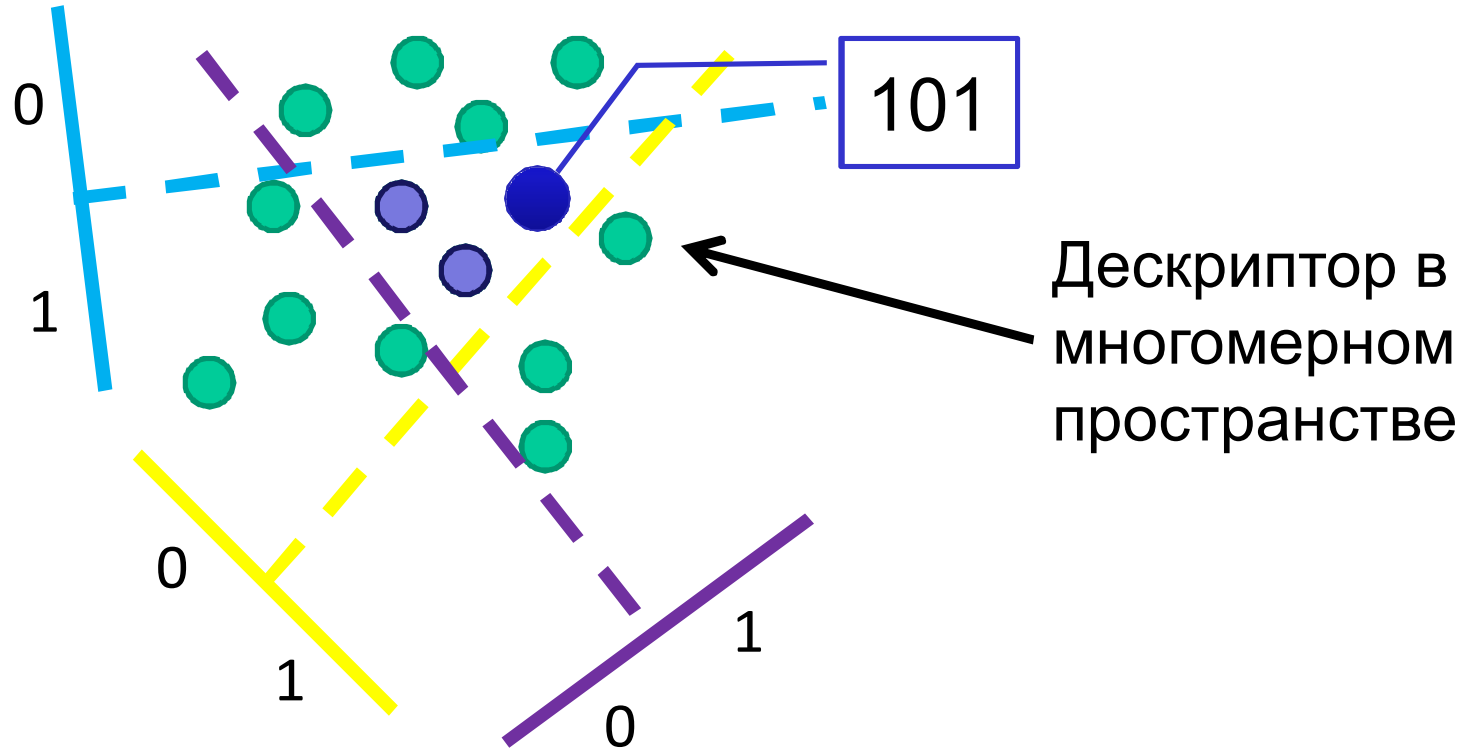
$$\Pr_{h \in \mathcal{F}} [h(\boldsymbol{x}) = h(\boldsymbol{y})] = sim(\boldsymbol{x}, \boldsymbol{y})$$

  - h(x) – семантическая хэш-функция

- Вариант формулировки:
  - $N_{100}(x)$ – ближайшие 100 в исходном пространстве по L2
  - Хотим найти h(x) – бинарную подпись, такую что
    - $N_{100}(x) = N_{100}(y)$
    - Где $N_{100}(y)$ – расстояние Хэмминга

Locality Sensitive Hashing (LSH)

- Возьмем случайную проекцию данных на прямую
- Случайно выберем порог, пометив проекции 0 или 1 (1 бит подписи)
- С увеличением числа бит подпись приближает L2-метрику в исходных дескрипторах



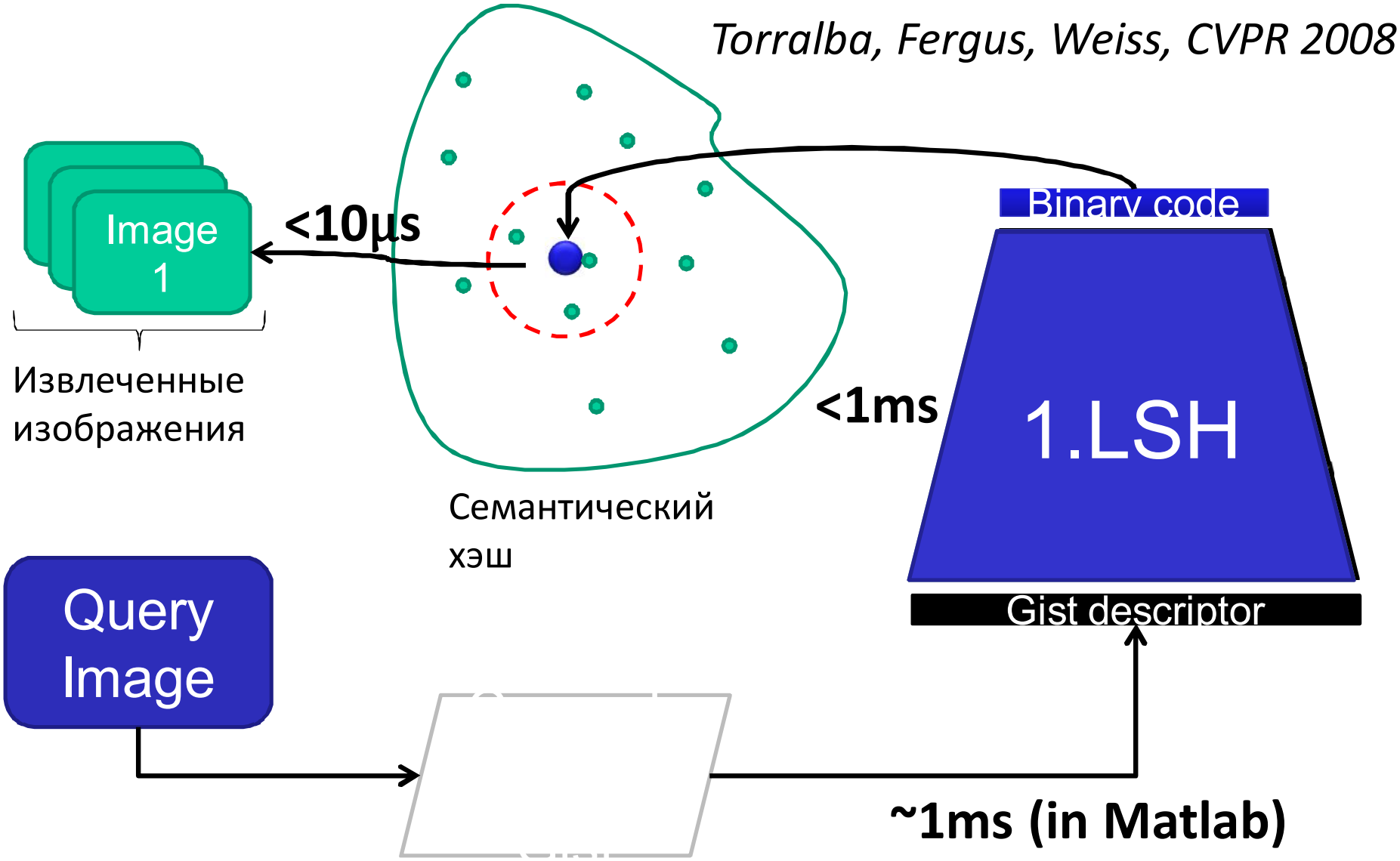
A. Andoni and P. Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In FOCS, 2006

Locality Sensitive Hashing (LSH)

Microsoft Research

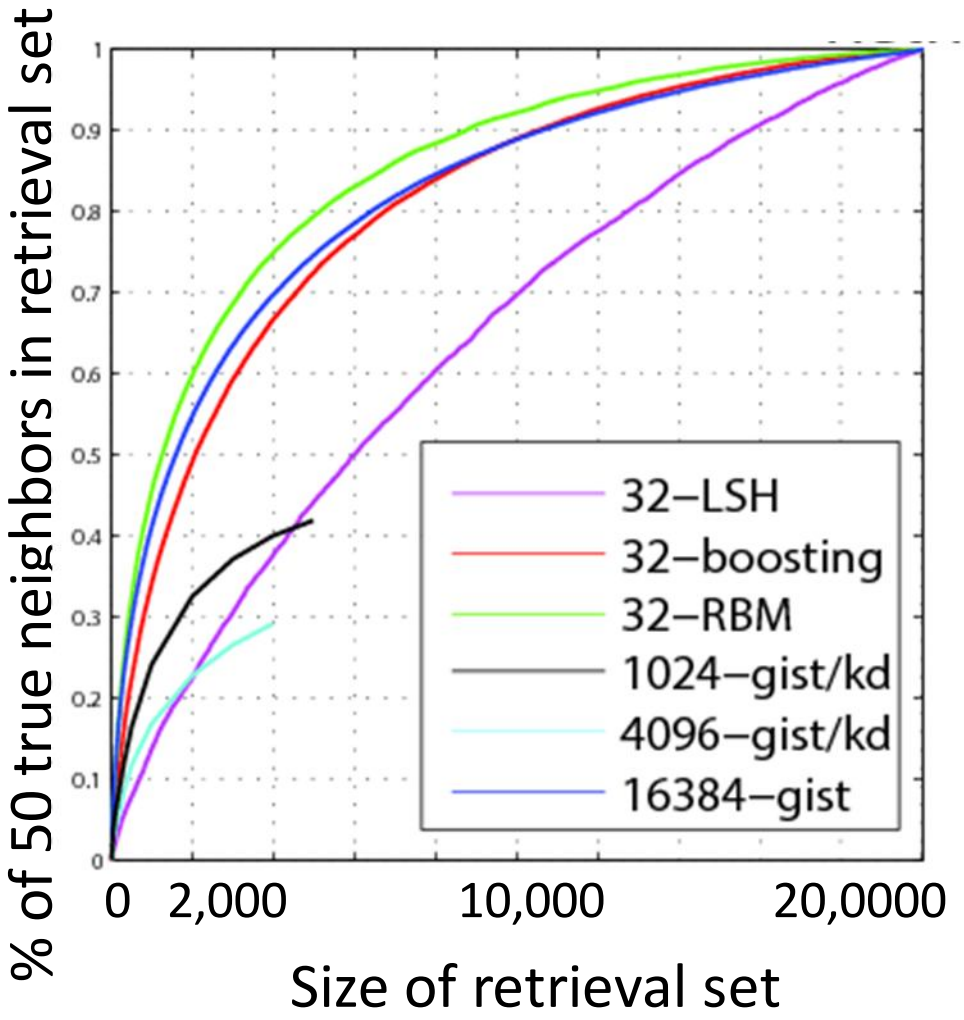
- Недостатки:
  - Приближение L2 лишь асимптотическое
  - На практике может потребоваться слишком много бит для подписи

Обучение бинарных кодов

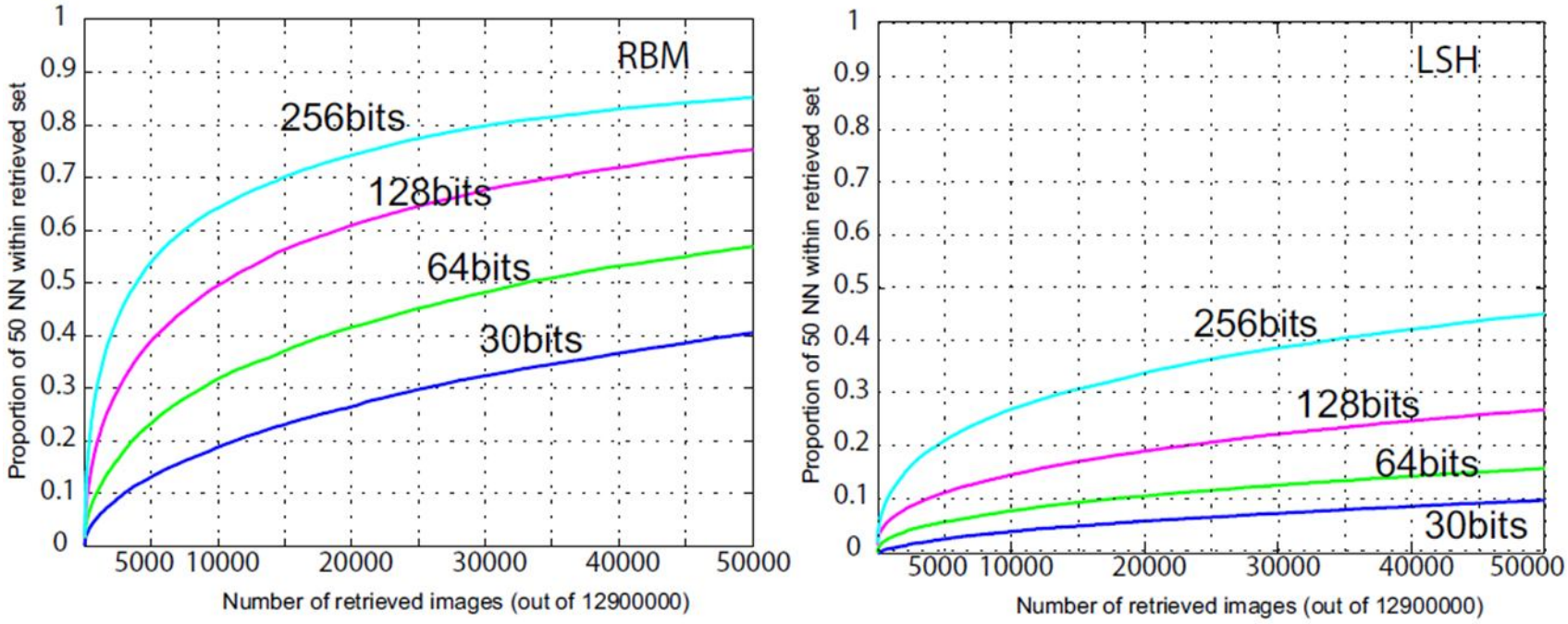


Сравнение на LabelMe

- 32-bit коды работают так же хорошо, как исходный дескриптор в 512 вещественных чисел
- Методы на основе обучения обгоняют LSH



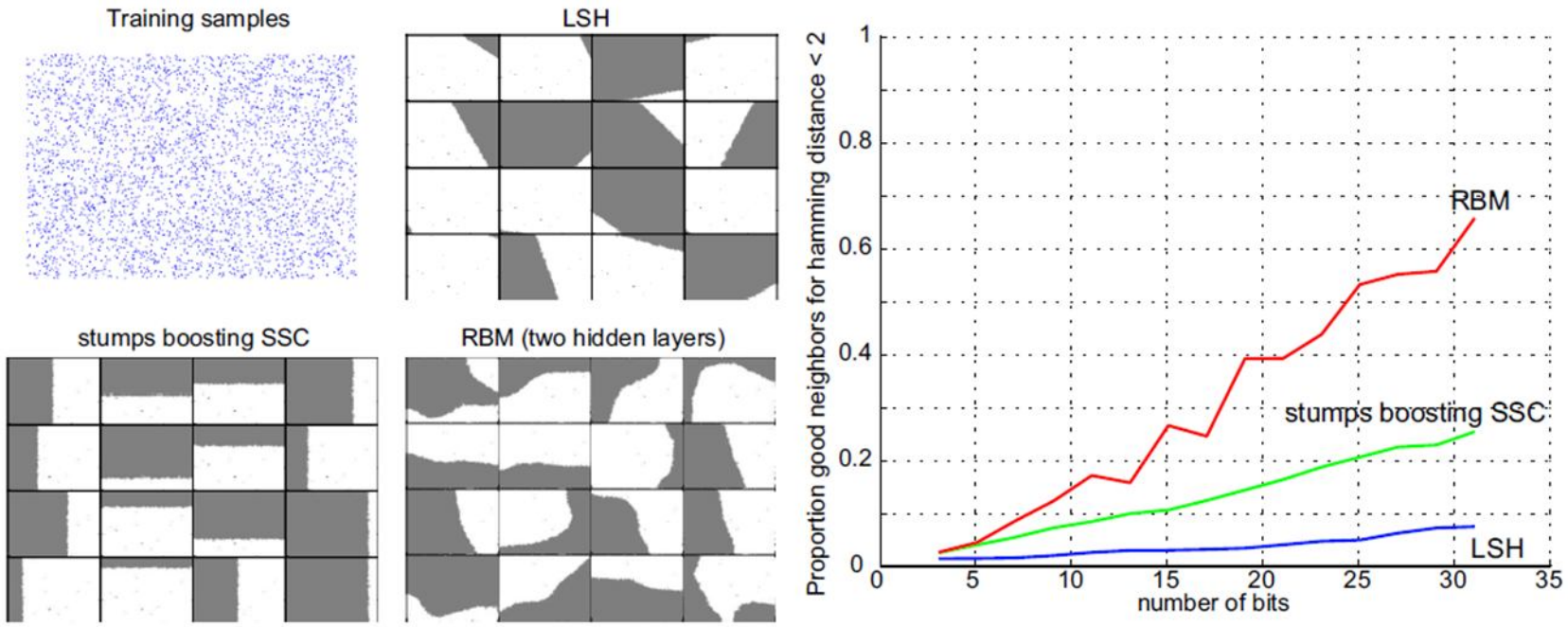
Сравнение на web



- Сравнение на 12.9M изображений

Сравнение

Анализ обучения



- Рассмотрим синтетический пример – точки на прямоугольнике
- Бинарные подписи фактически задают «базис» описания окрестности
- LSH даёт самое грубое описание окрестности

Microsoft Research

Спектральное хэширование

- Пусть x – исходные дескрипторы, y – подписи
- W – матрица расстояний между дескрипторами

$$W(i,j) = \exp(-\|x_i - x_j\|^2/\epsilon^2)$$

- Тогда задача оптимальных подписей:

$$minimize : \sum_{ij} W_{ij}\|y_i - y_j\|^2$$

$$subject\ to : y_i \in \{-1,1\}^k$$

$$\sum_i y_i = 0$$

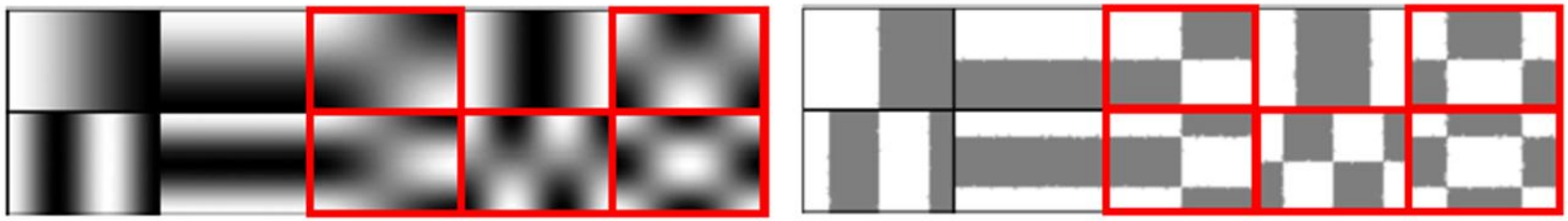
$$\frac{1}{n}\sum_i y_i y_i^T = I$$

- Ослабив постановку задачи получим эффективный алгоритм приближенного решения!

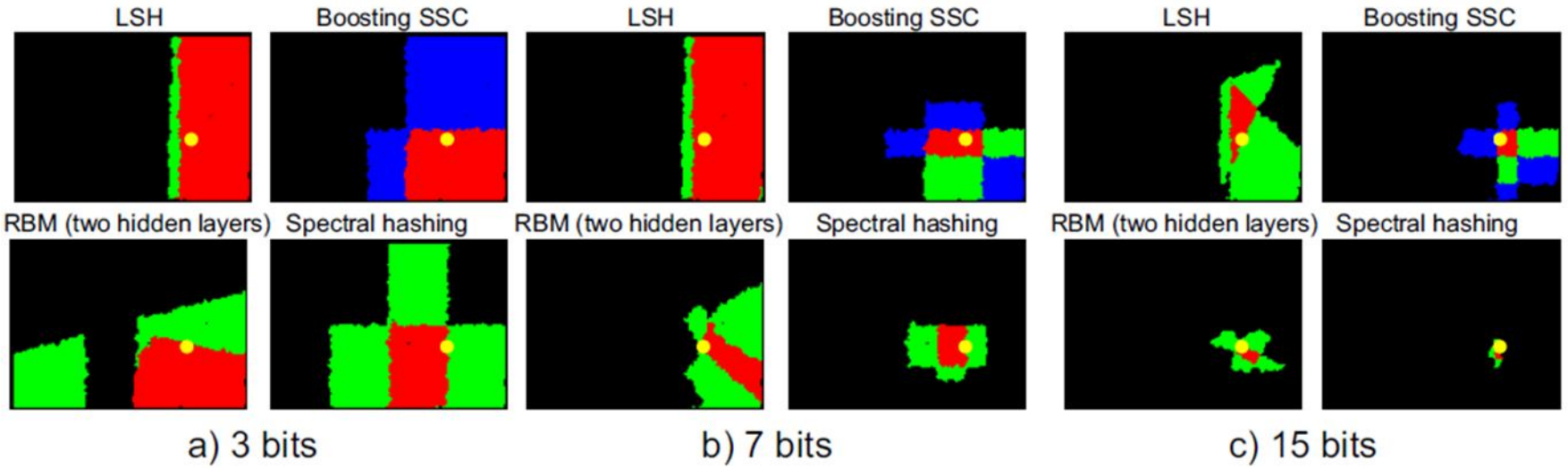
Weiss, Y., Torralba, A.B., Fergus, R.: Spectral hashing. NIPS, 2008

Спектральное хэширование

Microsoft Research



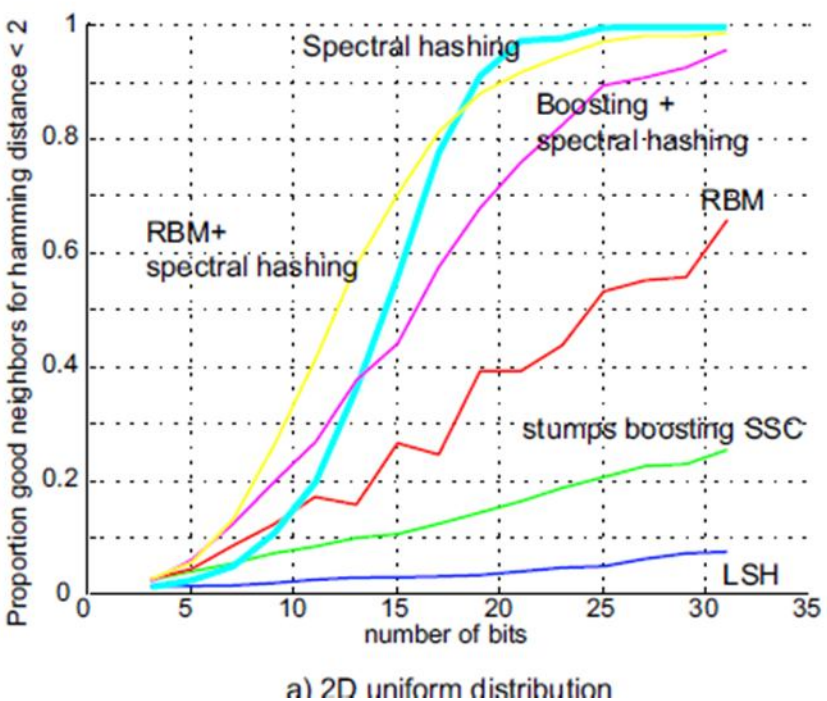
Построение базисных функций



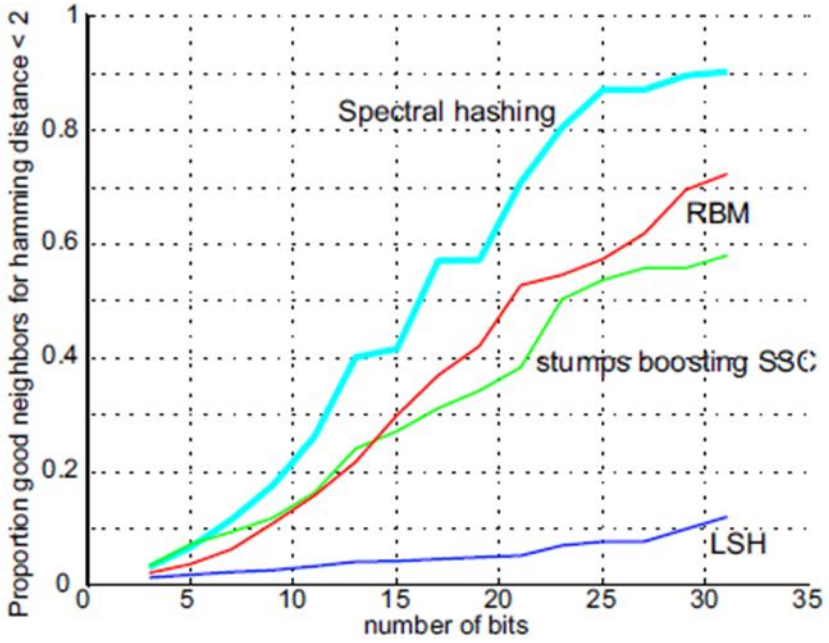
Сравнение окрестностей по функциям

Weiss, Y., Torralba, A.B., Fergus, R.: Spectral hashing. NIPS, 2008

Результаты



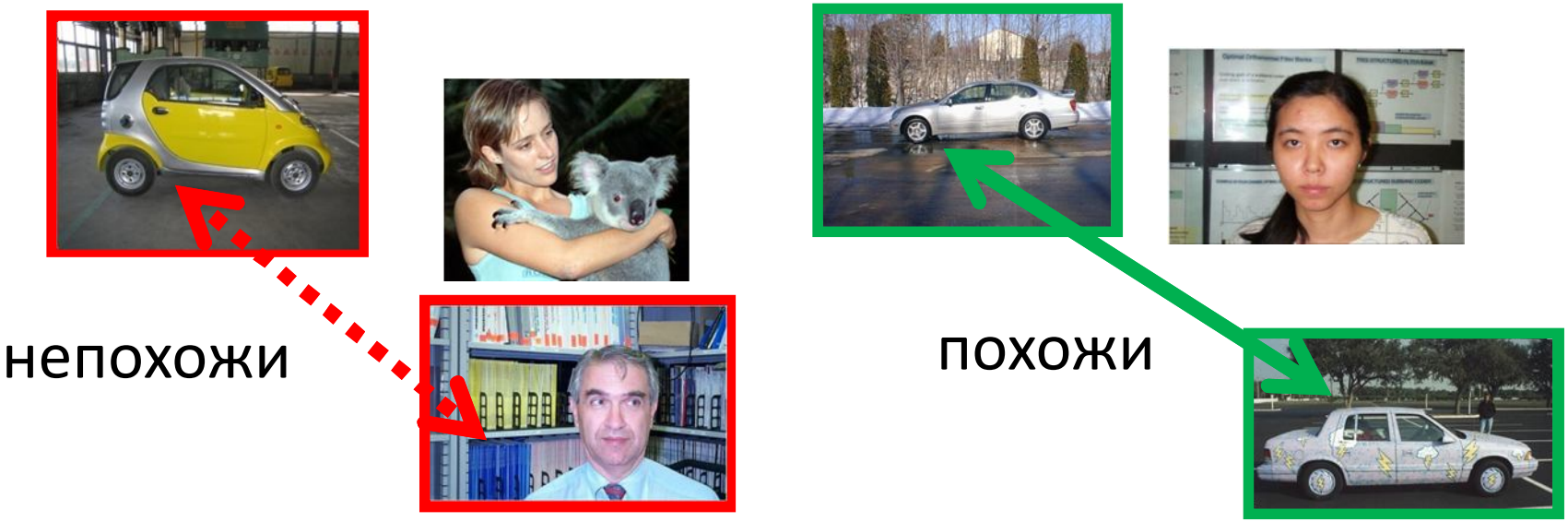
a) 2D uniform distribution



b) 10D uniform distribution

Weiss, Y., Torralba, A.B., Fergus, R.: Spectral hashing. NIPS, 2008

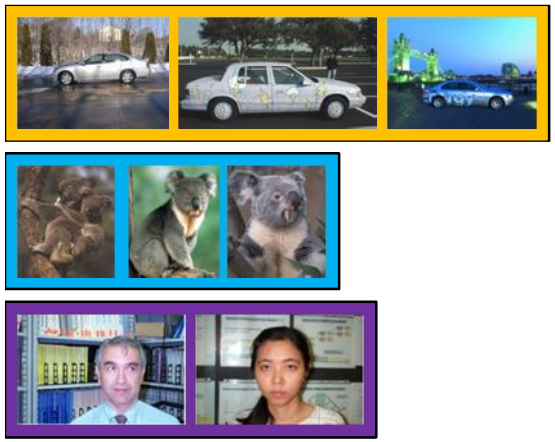
Обучаемые метрики



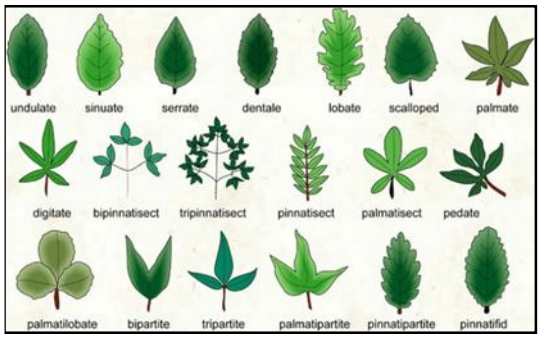
- Что если для наших дескрипторов Евклидово расстояние плохо подходит для описание близости изображений?
- Если можем выбрать похожие/непохожие изображения иначе, то можем обучить правильную метрику!
- Вид:
  - $r(x1,x2) = (x_1-x_2)^T A(x_1-x_2)$

Когда применимо?

Частично-размеченная база изображений



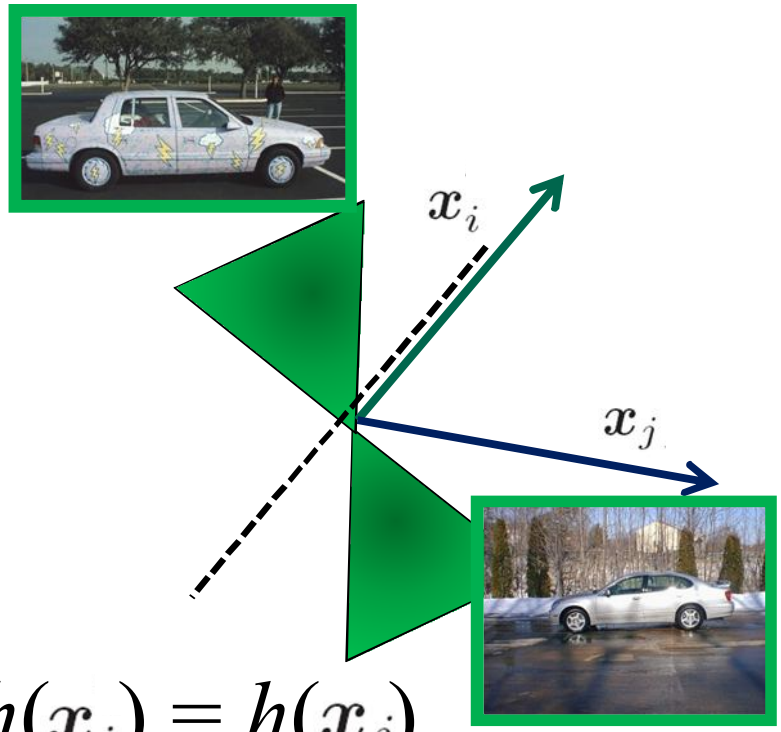
Полностью размеченная база



Специфическая проблемная область

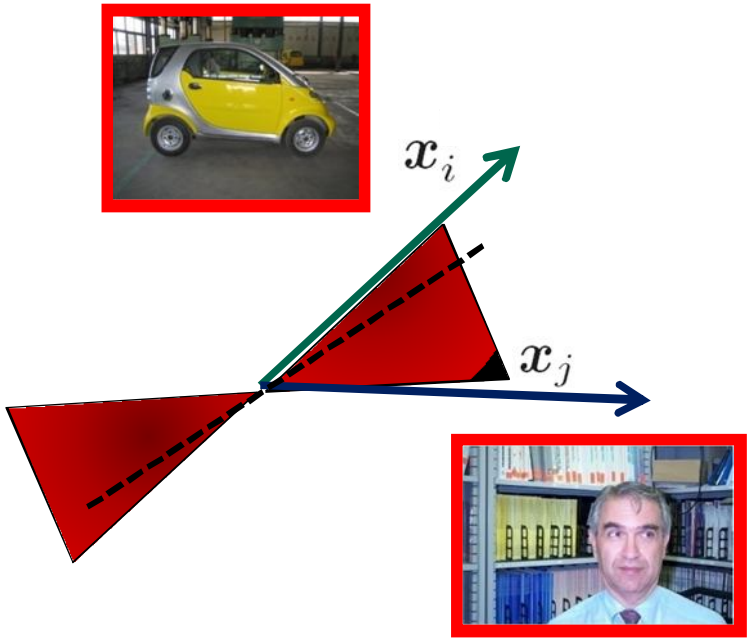
Быстрый поиск

Обучим расстояние через LSH



$$h(x_i) = h(x_j)$$

С меньшей вероятностью разбиваем подобные пары с ограничением сходства



$$h(x_i) \neq h(x_j)$$

С большей вероятностью разобьем пары с ограничением несходства

Результаты

- Сравнение на tiny images (80M)
- Обученная метрика позволяет найти те же результаты, но просмотрев меньше 1% базы
- Скорость – 0.5с вместо 45с

B. Kulis and K. Grauman. Kernelized locality-sensitive hashing for scalable image search. In *ICCV, 2009.*

Microsoft Research

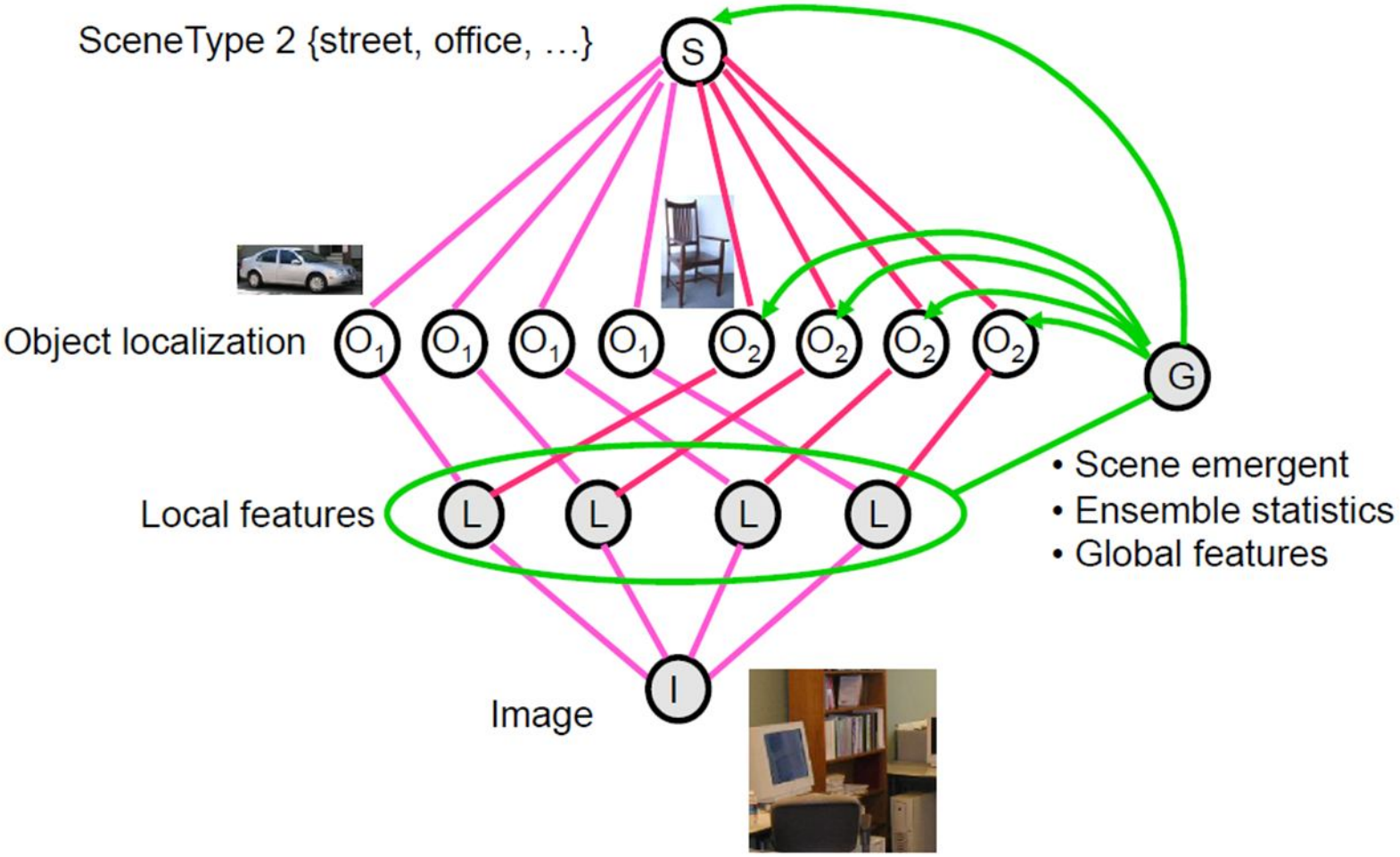
Резюме подписей

- Построение бинарных подписей с помощью хэш-функций позволяют существенно снизить размер индекса

- Сжимать можем любой дескриптор!

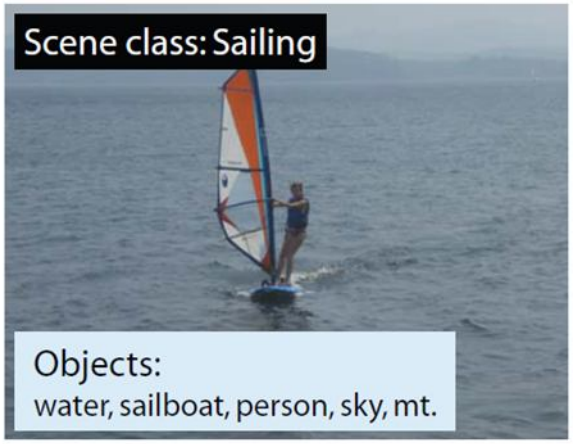
- Наилучшие результаты:
  - Если есть информация из внешних источников, какие изображения похожи, какие нет
    - Обучение метрики и бинарной подписи
  - Если такой информации нет
    - Спектральное хэширование

Модель сцены



- Вернёмся к исходной модели изображения

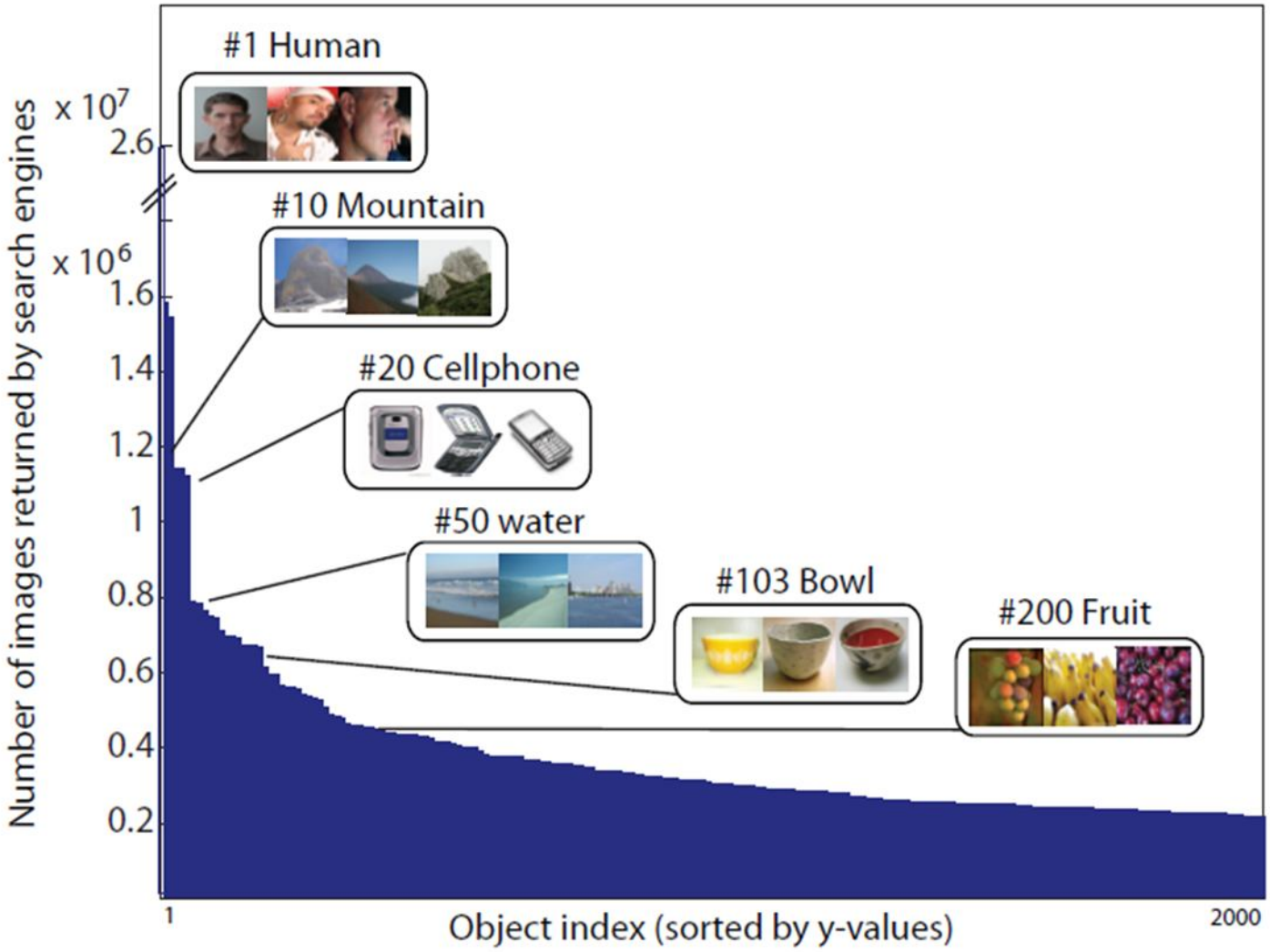
Описание сцены через объекты



- Если бы мы могли хорошо находить объекты и материалы (воду, небо, и т.д.), то можно было бы классифицировать и искать изображения по этим описаниям
- Но теперь мы можем!!!
  - HOG + SVM + части = детектор объектов
  - Сегментация + классификация = детектор материалов

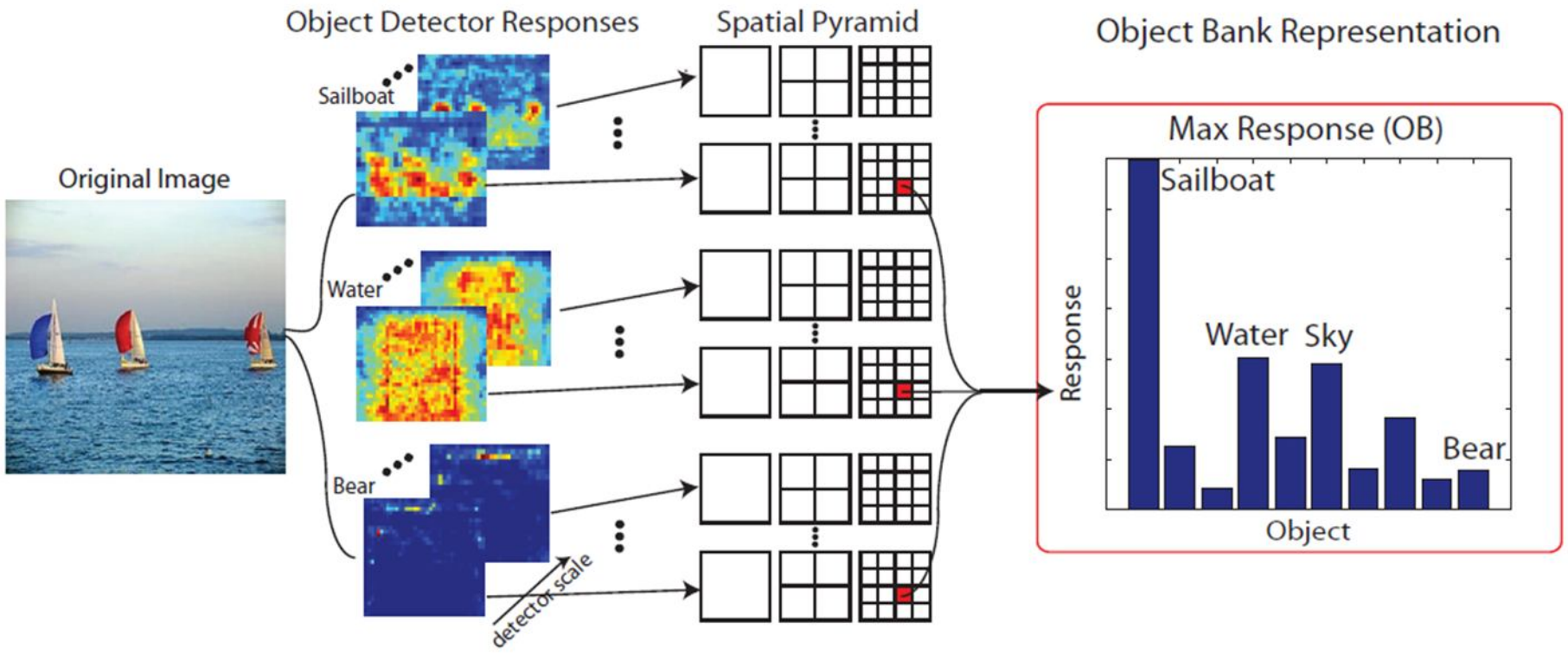
L.J. Li*, H. Su*, Y. Lim and L. Fei-Fei, Objects as Attributes for Scene Classification. ECCV 2010

Выбор объектов



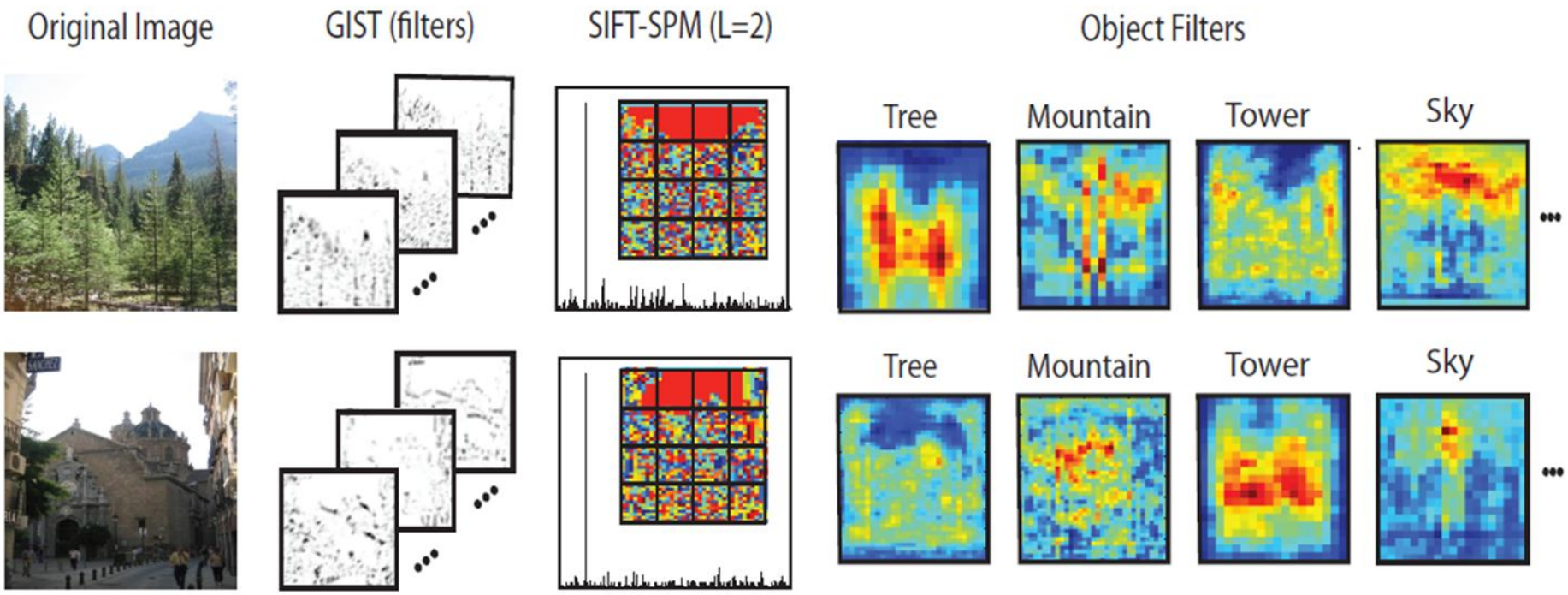
Число изображений, найденных по поиску через ключевые слова

Object filters



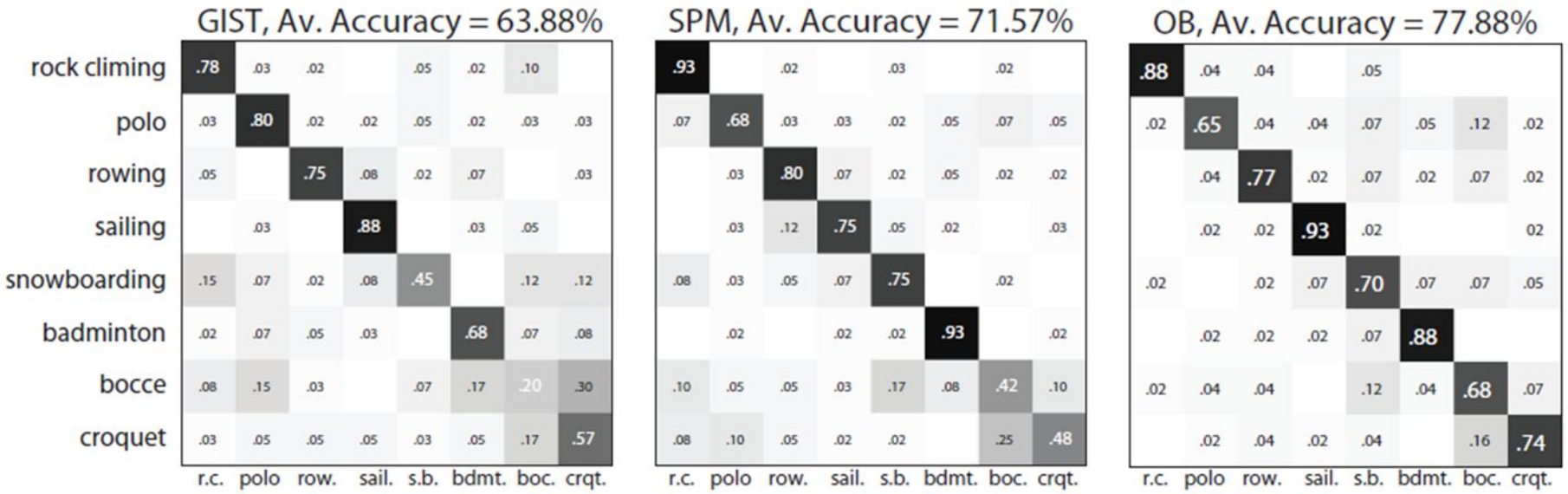
- «Фильтруем» изображения с помощью детекторов классов
- Суммируем области по прямоугольным областям
- Строим несколько уровней такого разрешения

Объединение признаков



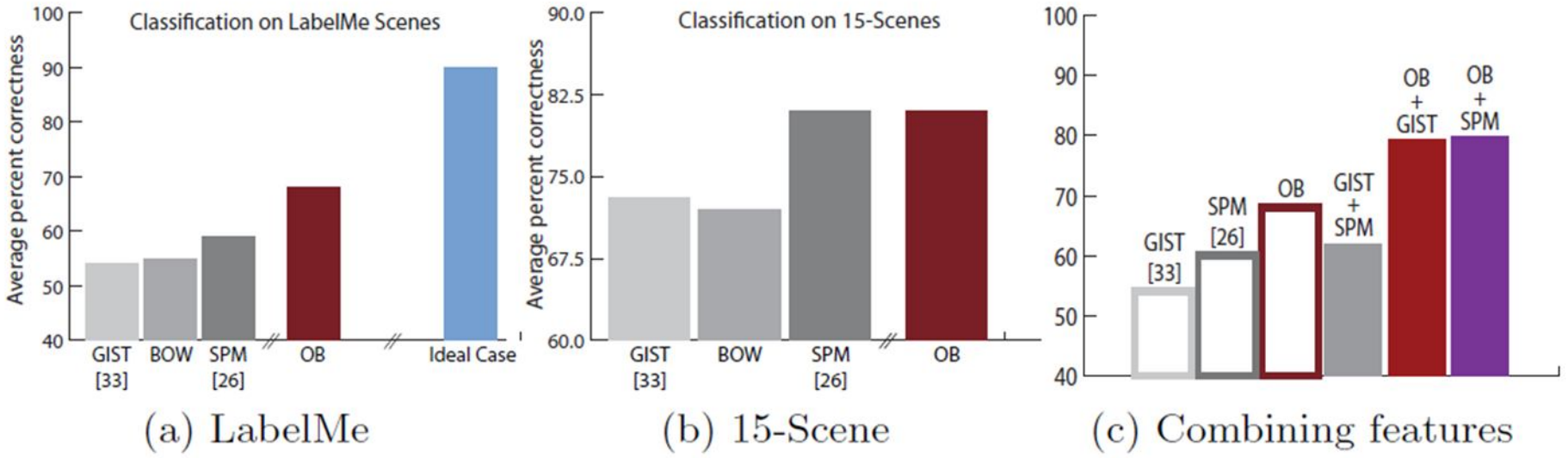
- Фильтры объектов, GIST и BOW дополняют друг друга, кодируя разный тип информации
- По всем признакам можно обучить линейный SVM для классификации изображений

Результаты



- Классификация изображений из базы UIUC-Sports
- Обучение на 70 изображениях, тестирование на 60

Результаты



- Object Filters дают результаты, сравнимые с лучшими признаками
- Комбинация признаков даёт state-of-art результаты

Microsoft Research

Смежные направления

- Классификаторы изображений как способ нелинейного хэширования
  - Lorenzo Torresani, Martin Szummer, Andrew Fitzgibbon. Efficient Object Category Recognition Using Classemes. *ECCV, 2010*.
  - Обучение 2500 классификаторов изображений
  - Квантование выходов, обучение линейного SVM на них
- Атрибуты изображений
  - Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. CVPR 2009
  - Сцену и объект можно описать через «прилагательные» (свойства объектов)
  - Соберём базу атрибутов изображений с помощью Mechanical Turk
  - Обучим классификаторы атрибутов

Microsoft Research

Резюме

- Разные признаки кодируют разные аспекты изображения
- Для преодоления семантического разрыва нужно использовать комбинацию признаков
- Детекторы объектов и базы изображения стали достаточно большими для использования как признаки
- Описание большой коллекции возможно только с очень небольшими бинарными подписями
- Подписи строятся по дескрипторам разными методами хэширования

Коды и базы

Microsoft Research

- Спектральное хеширование (Matlab)
  - http://www.cs.huji.ac.il/~yweiss/SpectralHashing/
- ImageNet
  - http://www.image-net.org/
- INRIA Holidays
  - http://lear.inrialpes.fr/people/jegou/data.php#holidays
- Oxford Buildings
  - http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/index.html
- UIUC Sports Event Dataset
  - http://vision.stanford.edu/lijiali/event_dataset/

Microsoft Research

На следующих лекциях

- Сегментация изображений
- Использование контекста и семантическая сегментация