

# Метод поиска нечетких дубликатов изображений на основе выявления точечных особенностей

© Пименов В.Ю.

Санкт-Петербургский Государственный университет,  
факультет Прикладной Математики-Процессов  
Управления  
vitaly.pimenov@gmail.com

## Аннотация

В настоящей работе предложен метод поиска нечетких дубликатов изображений на основе выявления и сравнения точечных особенностей, инвариантных к геометрическим, фотометрическим преобразованиям и изменению масштаба. Точечные особенности представляются дескрипторами типа PCA-SIFT.

Для решения задачи быстрой фильтрации множества дескрипторов предложена структура обратного индекса, использование которой позволяет получить значительный прирост производительности.

Экспериментальные результаты показывают эффективность применения предложенного метода для решения задачи поиска нечетких дубликатов изображений.

## 1. Введение

Целью настоящей работы являлась разработка программного комплекса, реализующего метод поиска нечетких дубликатов изображений, оценка производительности метода и анализ факторов, оказывающих влияние на производительность метода.

Поиск нечетких дубликатов изображений является одной из ключевых задач машинного зрения [1] и в настоящее время привлекает внимание исследователей в области информационного поиска. Это связано с тем, что использование информации о

нечетких дубликатах позволяет разрабатывать эффективные алгоритмы для решения задач поиска новостей [2], определения и отслеживания новостных сюжетов [3], поиска нарушений авторского права [4].

В задачах поиска новостей использование информации о нечетких дубликатах позволяет отслеживать новости, посвященные одному событию, среди источников на разных языках и позволяет повысить производительность методов на 10-20% [3].

**Определение.** Два изображения, которые могут быть переведены друг в друга путем элементарных преобразований, таких как, поворот, сдвиг, изменение угла обзора, изменение разрешения, изменение масштаба, изменение освещения, называются *нечеткими дубликатами*.

Естественными нечеткими дубликатами являются фотоснимки и видеок cadры одной сцены, сделанные с различных ракурсов, при различном освещении. Кроме того нечеткие дубликаты возникают в результате редактирования изображений и при создании коллажей.

Внимание к задаче поиска нечетких дубликатов возросло с возникновением в 2003 г. семинара TRECVID [5]. Обзор ряда результатов представлен в работе [2].

Сложность задачи поиска нечетких дубликатов обусловлена необходимостью представления изображений в формате, обеспечивающем инвариантность по отношению к потенциально сложным геометрическим и фотометрическим преобразованиям.

Задачи поиска нечетких дубликатов включают следующие частные задачи: поиск дубликатов по запросу, сравнение пары изображений, кластеризация коллекции изображений. В настоящей работе рассматривалась задача кластеризации.

Процесс решения задачи состоял из двух основных этапов:

- 1) построение векторной модели изображения;
- 2) проведение кластеризации.

Для построения векторной модели изображения могут быть использованы глобальные или локальные свойства изображения.

Глобальные свойства изображения, в общем случае, не гарантируют эффективности идентификации нечетких дубликатов. Так, использование цветовых свойств, к примеру матрицы изменения яркости, может приводить к ошибкам в случае изменения условий освещения сцены или использования ряда эффектов [8].

С другой стороны, использование характерных локальных свойств, устойчивых к геометрическим и фотометрическим преобразованиям, представляется более предпочтительной

альтернативой. Одним из подтверждений успешности данного подхода является *метод выявления точечных особенностей*, предложенный в работах D. Lowe (см., например [1]). Точечные особенности (local interest points, перевод заимствован из работы [7]) являются выделяющимися областями изображений, выявленными в различных масштабах.

Для выявления точечных особенностей разработаны различные методы-детекторы. Обзор и сравнение этих методов представлены в работе [1]. В настоящей работе использован метод разности гауссианов (Difference of Gaussian) [1], с помощью которого выявляются точечные особенности, инвариантные к изменению масштаба и аффинным преобразованиям. Для описания точечных особенностей использованы дескрипторы PCA-SIFT [8], устойчивые к фотометрическим преобразованиям.

Для проведения кластеризации использовался модифицированный алгоритм  $QT$  [9]. Этот выбор обоснован простотой программной реализации. Альтернативные решения: locality sensitive hashing [10] и методы, использующие теорию графов [11].

Атомарная операция алгоритма  $QT$  – вычисление расстояния между парой векторов. Алгоритм обладает квадратичной сложностью по атомарной операции. В связи с этим в работе предложен метод построения индексной структуры для ускорения вычисления расстояния. Однако, разработка метода кластеризации не входила в число задач настоящей работы.

## 2. Алгоритм выявления точечных особенностей

В работе использован алгоритм выявления точечных особенностей, предложенный в [1]. Ряд последующих исследований подтверждают эффективность данного подхода (см., например [3,6,8]).

Критерием, на основе которого выявляются точечные особенности изображения, является инвариантность к изменению масштаба.

Основной процедурой алгоритма является построение масштабируемого пространства (scale space) [12,13,14].

**Определение [14].** Построение масштабируемого пространства представляет собой операцию разложения исходного сигнала  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  в семейство  $\{T_t f | t \geq 0\}$  постепенно сглаживаемых, упрощающихся версий сигнала, удовлетворяющих ряду аксиом [14],

среди которых в контексте рассматриваемой задачи важна инвариантность к изменению масштаба.

Доказано [15], что единственным непрерывным линейным масштабируемым пространством является Гауссово масштабируемое пространство, для которого

$$T_t f = K_{\sqrt{2t}} * f,$$

где  $K_{\sigma}$  - функция Гаусса со стандартным отклонением  $\sigma$ .

Для выявления точечных особенностей вычисляется разность гауссианов

$$D(f) = K_{k\sigma} * f - K_{\sigma} * f,$$

где  $k$  – постоянная величина.

Выбор такой функции в качестве детектора точечных особенностей обусловлен следующими причинами.

Разность гауссианов является аппроксимацией масштабно-нормализованного лапласиана гауссиана,  $\sigma^2 \nabla^2 K_{\sigma}$ , как показано в работе [1]. На основе экспериментальных данных, полученных в работе [16], можно утверждать, что экстремумы функции  $\sigma^2 \nabla^2 K_{\sigma}$  являются наиболее стабильными точечными особенностями изображений, инвариантными по отношению к изменению масштаба, в сравнении в экстремумами других функций, использовавшихся в качестве детекторов: градиента, гессиана, угловой функции Харриса.

Взаимосвязь функций можно проиллюстрировать исходя из уравнения теплопроводности

$$\frac{\partial K_{\sigma}}{\partial \sigma} = \sigma \nabla^2 K_{\sigma},$$

которое после замены частной производной на конечную разность, приобретает вид

$$\frac{K_{k\sigma} - K_{\sigma}}{k\sigma - \sigma} \approx \sigma \nabla^2 K_{\sigma}.$$

Нетрудно заметить, что в числителе, после этого преобразования, оказывается разность гауссианов. В конечном итоге получаем приближенное выражение для масштабно-нормализованного лапласиана гауссиана

$$\sigma^2 \nabla^2 K_{\sigma} \approx \frac{1}{(k-1)} (K_{k\sigma} - K_{\sigma}).$$

Так как коэффициент  $k-1$  является постоянной величиной, то он не влияет на расположение экстремумов. Экспериментальные результаты, полученные в работах [1,4,6,7,8] подтверждают эффективность применения подобной аппроксимации для выявления точечных особенностей.

Кроме того, использование разности гауссианов не требует дополнительных вычислений: после построения масштабируемого пространства, для вычисления разности гауссианов требуется лишь произвести вычитания.

После расчета разностей гауссианов производится поиск экстремумов. Экстремумы определяются путем сравнения яркости точек с соседними. После этого отсеиваются точки, имеющие невысокие контраст, так как они чувствительны к возможному шуму. Кроме того, отсеиваются точки, расположенные на границах. Детальное изложение алгоритмов, используемых при проведении этих операций можно найти в работе [1].

После нахождения экстремумов, инвариантных к геометрическим преобразованиям, необходимо представить их в форме, инвариантной к фотометрическим преобразованиям. Для этого вычисляются дескрипторы точечных особенностей. В настоящей работе использованы дескрипторы PCA-SIFT. Их расчет включает два этапа.

На первом этапе рассчитывается дескриптор SIFT (Scale Invariant Feature Transform) [1]. Дескриптор этого типа представляет собой 128-мерный вектор, сформированный из значений ориентаций градиентов, вычисленных для области, окружающей точечную особенность.

Для построения дескриптора точечной особенности вычисляются ориентации и величины градиентов изображения соответствующего масштаба в области размером  $4 \times 4$  пиксела вокруг особенности. Для обеспечения инвариантности относительно изменения ориентации производится преобразование локальных координат. Кроме этого, величины градиентов взвешиваются с помощью гауссовской весовой функции для того, чтобы акцентировать центр области – точечную особенность, для которой рассчитывается дескриптор. Для того, чтобы исключить влияние нелинейных фотометрических преобразований, градиенты, величины которых превосходит

экспериментально определенное значение 0.2 [1], исключаются из дальнейшей обработки.

После этого ориентации оставшихся градиентов распределяются по дискретной таблице из восьми ячеек, и таким образом формируется вектор-descriptor из  $4 \times 4 \times 8 = 128$  элементов.

Обоснование применимости такого типа descriptor и более детальное изложение алгоритма его расчет приведены в работе [1].

На втором этапе вычисленный descriptor SIFT подвергается процедуре редукции размерности с помощью метода главных компонент (Principal Component Analysis, PCA). Это подход, предложенный в работе [8], и успешно примененный в работах [6,8], позволяет сократить вычислительную сложность алгоритмов сравнения descriptor за счет снижения размерности descriptor со 128 до 36, не теряя при этом в точности.

После вычисления descriptor PCA-SIFT изображения больше не используются, все дальнейшие операции производятся над 36-мерными векторами-descriptorами.

### 3. Алгоритмы сравнения и фильтрации

Для проведения кластеризации был использован алгоритм *QT*. Атомарная операция этого алгоритма – вычисление расстояния между двумя векторами.

В рассматриваемой задаче нет однозначного подхода к введению расстояния между изображениями. Это связано с тем, что различным изображениям соответствуют различные количества descriptorов, что затрудняет введение единообразной векторной модели изображения.

В связи с этим в настоящей работе использовано решение, основанное на предложенном в работе [6] алгоритме взаимнооднозначного сопоставления точечных особенностей.

Это решение заключается в том, что расстояние между изображениями рассчитывается как

$$\rho(I_n, I_m) = \frac{1}{\|OOS(I_n, I_m)\|}, \quad (1)$$

где  $OOS(I_n, I_m)$  – множество пар точечных особенностей, соответствующих рассматриваемым изображениям, которые являются ближайшими соседями друг друга. Это означает, что пара точечных особенностей  $(P, Q)$  будет включена в множество

$OOS(I_n, I_m)$  тогда и только тогда, когда  $P \rightarrow Q$  и  $Q \rightarrow P$ , где  $\rightarrow$  обозначает отношение однозначного ближайшего соседства, т.е.

$$\rho(P, Q) < \rho(P, Q'), \quad \forall Q' \in I_m,$$

$$\rho(Q, P) < \rho(Q, P'), \quad \forall P' \in I_n,$$

где  $\rho(P, Q)$  – евклидово расстояние между 36-мерными векторами-дескрипторами  $P$  и  $Q$ . Использование такого расстояния возможно, так как результатом применения метода главных компонент являются элементы векторного пространства с ортогональным базисом.

Использование такого расстояния позволяет сократить число неверных совпадений между дескрипторами, хотя и не исключает их возникновения.

В работе [6] представлена подробная аргументация, обосновывающая преимущества использования симметричного взаимнооднозначного соответствия по сравнению с другими подходами: соответствием многие-ко-многим, многие-к-одному, один-ко-многим, один-к-одному без симметрии.

Таким образом, в центре атомарной операции алгоритма  $QT$  находится процедура построения множества  $OOS(I_n, I_m)$ . В свою очередь эта процедура основана на процедуре поиска для данного дескриптора точечной особенности  $P \in I_n$  ближайшего соседа  $Q \in I_m$ . Поскольку число точечных особенностей, выявленных в одном изображении может достигать нескольких тысяч, процедура поиска ближайшего соседа может оказаться вычислительно сложной.

Для понижения вычислительной сложности прибегают к использованию различных способов предварительной фильтрации множества дескрипторов. В числе этих способов многомерные индексные структуры, в частности locality sensitive hashing [10], использованная в работе [4] и LIP-IS (local interest point index structure), предложенная в работе [6], а также методы разделения пространства, в частности KD-деревья, использованные в работе [17].

В настоящей работе предлагается новая индексная структура, основанная на структуре LIP-IS. Основная концепция, заимствованная из LIP-IS, заключается в нормализации и дискретном хешировании дескрипторов.

Оригинальная индексная структура LIP-IS, предложенная в работе [6], представляет собой таблицу, в которой каждому из 36 измерений вектора дескриптора, которое предварительно приводится в диапазон  $[-1, 1]$ , соответствуют 8 ячеек, с разрешением  $\Delta = 0.25$ . Таким образом, для дескриптора  $P = (p_1, \dots, p_{36})$ , координата  $p_i$  хешируется по правилу

$$H(p_i) = \left\lfloor \frac{p_i + 1}{\Delta} \right\rfloor.$$

Процедура фильтрации заключается в том, что после индексирования всех дескрипторов изображения  $I_m$ , для сравнения с дескриптором изображения  $I_n$  из индексной структуры выбираются лишь те дескрипторы, которые удовлетворяют следующему условию:

$$Sim(P, Q) = \sum_{i=1}^{36} C(p_i, q_i) \geq 36 - M,$$

где

$$C(q_i, p_i) = \begin{cases} 1, & \text{если } |H(q_i) - H(p_i)| \leq 1 \\ 0, & \text{в противном случае} \end{cases}.$$

Здесь,  $M$  – параметр, определяющий пропускную способность. Его значение может быть вычислено, исходя из значения  $\Delta$  [6], однако, в настоящей работе значение  $M$  полагалось равным нулю.

Фильтрация, осуществляемая подобным способом, может быть реализована на основе простейших битовых операций.

Однако, для осуществления фильтрации необходим линейный обход всех дескрипторов, и таким образом, сравнение двух изображений является квадратичным алгоритмом по операции фильтрации.

Для сокращения вычислительных затрат в настоящей работе предложено использование обратного индекса для предварительной фильтрации и уменьшения, тем самым, размера линейного обхода, используемого для проведения основной фильтрации.

Обратный индекс представляет собой структуру, осуществляющую отображение

$$Q \rightarrow \{Q' | C(q_1, q'_1) = 1, C(q_2, q'_2) = 1, C(q_3, q'_3) = 1\}.$$

Таким образом, предварительная фильтрация состоит в том, что на основе обратного индекса определяется множество дескрипторов,



для которых первые 3 измерения удовлетворяют условию основной фильтрации.

Обратный индекс строится один раз для изображения. Операция получения дескрипторов из обратного индекса не требует обхода, в отличие от основной фильтрации. Так как для построения обратного индекса используются лишь 3 измерения, то его использование не приводит к существенному расходу памяти: для каждого изображения необходимо хранить лишь таблицу из 729 столбцов. Согласно экспериментальным результатам, использование обратного индекса позволяет сократить число дескрипторов, участвующих в линейном обходе основной фильтрации на 40-80%. Использование более трех измерений для построения обратного индекса не оправдано, так как не приводит к более существенному сокращению числа дескрипторов, однако при этом требует хранения таблицы из 6561 столбцов.

Таким образом, процедура построения множества  $OOS(I_n, I_m)$  может быть представлена следующими этапами:

- 1) Хеширование координат дескрипторов изображений.
- 2) Построение обратных индексов для изображений.
- 3) Для каждого дескриптора  $P \in I_n$  :
  - i) проведение предварительной фильтрации дескрипторов изображения  $I_m$  в соответствии с дескриптором  $P \in I_n$  ;
  - ii) проведение основной фильтрации дескрипторов изображения  $I_m$  в соответствии с дескриптором  $P \in I_n$  ;
  - iii) вычисление евклидова расстояния  $\rho(P, Q')$  для каждого дескриптора  $Q' \in I_m$  , удовлетворяющего условиям фильтрации;
  - iv) определение дескрипторов  $\{Q^*\} \subset I_m$  – ближайших соседей для  $P \in I_n$  ;
  - v) если ближайший сосед  $Q^*$  единственный:
    - a) проведение предварительной фильтрации дескрипторов изображения  $I_n$  в соответствии с дескриптором  $Q^*$  ;
    - b) проведение основной фильтрации дескрипторов изображения  $I_n$  в соответствии с дескриптором  $Q^*$  ;

- c) вычисление евклидова расстояния  $\rho(P', Q^*)$  для каждого дескриптора  $P' \in I_m$ , удовлетворяющего условиям фильтрации;
- d) определение дескрипторов  $\{P^*\} \subset I_n$  – ближайших соседей для  $Q^*$ ;
- e) если ближайший сосед  $P^*$  единственный и  $P^* \equiv P$ , то множеству  $OOS(I_n, I_m)$  добавляется пара  $(P^*, Q^*)$ .

После этого процедура кластеризации с помощью алгоритма  $QT$  может быть представлена следующими этапами:

- 1) Определение максимального диаметра кластера.
- 2) Выбор изображения  $I$  из коллекции.
- 3) Для каждого изображений  $I'$  из коллекции, отличного от  $I$ :
  - a) построение множества  $OOS(I, I')$ ;
  - b) вычисление расстояние в соответствии в формулой (1);
  - c) если расстояние меньше максимального диаметра, изображение  $I'$  добавляется к кластеру с центром  $I$ .
- 4) Удаление всех изображений, добавленных к кластеру с центром  $I$  из коллекции.
- 5) Переход к шагу 2).

При проведении экспериментов значение максимального диаметра кластера полагалось равным 0,2, т.е. изображения  $I'$ , добавляемые в кластер с центром  $I$ , должны иметь по крайней мере 5 пар дескрипторов в множестве  $OOS(I, I')$ .

Использованный алгоритм не допускает попадания одного изображения в несколько различных кластеров.

#### 4. Программная реализация

Предложенный метод поиска нечетких дубликатов изображений был реализован в виде набора программных модулей, разработанных на языке Java. Этот выбор обусловлен удобством разработки и кроссплатформенностью языка. Для тестирования результатов работы метода использовались свободно доступные библиотеки, предназначенные для решения схожих задач [18,19], и их исходные коды.

## 5. Постановка эксперимента

Для оценки производительности разработанного метода был проведен эксперимент. Для проведения эксперимента была использована коллекция изображений предоставленная организаторами семинара РОМИП. Коллекция состояла из 37925 изображений различного размера и качества. Количество точечных особенностей для одного изображения варьировалось от 0 до 11985. В среднем на одно изображение приходилось 652 точечные особенности. Следует отметить, что это вдвое меньше, чем в коллекции TRECVID 2003 [5], которая использовалась при проведении экспериментов в работе [6].

Кластеризация достаточно большой коллекции изображений – вычислительно трудная задача. Поэтому при проведении эксперимента коллекция разделялась на части, которые обрабатывались в параллельном режиме. Это было возможно, так как выбранный для кластеризации алгоритм *QT* допускает легко реализуемое распараллеливание.

Для оценки результатов эксперимента вычислялись значения четырех метрик: уровень ошибок первого рода, уровень ошибок второго рода, точность и полнота. Оценка проводилась организаторами семинара РОМИП. Методология оценки и описания метрик описаны в [20].

Кроме этого, отдельно были проведены измерения производительности метода. Измерения производительности проводились на компьютере с процессором Intel Core 2 Duo 1.83ГГц с объемом оперативной памяти 2Гб.

## 6. Анализ результатов эксперимента

После проведения процедуры оценки были получены следующие значения метрик:

- уровень ошибок первого рода:  $2.45 \times 10^{-4}$ ;
- уровень ошибок второго рода: 0.36;
- точность: 0.69;
- полнота: 0.63.

Полученные результаты в целом согласуются с результатами, полученными в работе [6] при использовании схожего метода поиска нечетких дубликатов.

Относительно низкое значение точности объясняется использованием большого радиуса кластера в алгоритме *QT*: для

попадания в кластер изображение среди своих точечных особенностей должно было иметь не менее пяти пар симметрично ближайших соседей с точечными особенностями центра кластера. Это составляет 0.76% от среднего числа точечных особенностей на изображение.

Увеличение порога включения в кластер позволило бы существенно выиграть в точности, однако это привело бы к потере в полноте. Это объясняется тем, что для 20% всех изображений было выявлено менее 100 точечных особенностей. Подобные низкие показатели связаны с низким качеством изображений.

Кроме того, важным фактором, оказывающим влияние на точность кластеризации, является шаг дискретизации  $\Delta$ , используемый при хешировании координат дескрипторов при их индексировании. Величина шага была выбрана равной 0.25, также как в работе [6]. Отказ от хеширования позволяет повысить точность более чем на 10% [6], однако это приводит к существенному росту вычислительной нагрузки; в настоящей работе не проводились эксперименты без использования хеширования и индексирования, результаты, представленные в [6], показывают потерю производительности на один порядок.

Измерения производительности показали следующий результат: сравнение двух изображений со средним числом точечных особенностей 792 на изображение занимает 0.009-0.011 секунды. Результат, полученный в работе [6] составляет 0.028 секунды. С поправкой на то, что среднее число точечных особенностей в изображениях коллекции TRECVID 2003 [5], использованной в работе [6], составляет 1200, получаем, что за счет введения предварительной фильтрации производительность возросла в 1.86 раза.

## 7. Заключение

В настоящей работе предложен метод поиска нечетких дубликатов изображений на основе выявления и сравнения точечных особенностей инвариантных к геометрическим, фотометрическим преобразованиям и изменению масштаба. Подобные методы сравнительно недавно стали получать применение в задачах информационного поиска, и на сегодняшний день, привлекают внимание все большего числа исследователей. Это связано с тем, что методы, основанные на выявлении точечных особенностей, и в частности, основанные на теории масштабируемого пространства,

обладают большим потенциалом в применении к задачам поиска новостей и мультимедийной информации, имеют биологическое обоснование [1,15] и согласуются с моделями обработки сигналов зрительными системами живых организмов [15]. Экспериментальные результаты, полученные в настоящей работе и ряде предшествующих работ [1,2,3,4,6,7,8] показывают высокую эффективность применения методов этого класса для решения задач поиска нечетких дубликатов изображений.

В числе направлений дальнейшего развития полученных результатов наиболее значимы следующие:

- повышение производительности метода для обеспечения возможности его применения в системах реального времени;
- разработка специализированных методов кластеризации, учитывающих специфику используемых моделей изображения.

## Литература

- [1] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. In *International Journal of Computer Vision*, volume 60, pages 91-110, 2004.
- [2] S.F. Chang, W. Hsu, L. Kennedy, L. Xie, A. Yanagawa, E. Zavesky and D.-Q. Zhang. Columbia University TRECVID-2005 Video Search and High-Level Feature Extraction. In *TRECVID*, 2005.
- [3] X. Wu, C.-W. Ngo and Q. Li. Threading and Autodocumenting News Videos. In *Signal Processing Magazine*, 2006.
- [4] Y. Ke, R. Suthanakar and L. Huston. Efficient Near-Duplicate Detection and Sub-Image Retrieval. In *ACM Multimedia Conference 2004*, pages 869-876, 2004.
- [5] TREC Video Retrieval Evaluation (TRECVID) Web Site. <http://www-nlpir.nist.gov/projects/trecvid>
- [6] W.-L. Zhao, C.-W. Ngo, H.-K. Tan and X. Wu. Near-Duplicate Keypoint Identification with Interest Point Matching and Pattern Learning. In *IEEE Transactions on Multimedia*, volume 9(5), pages 1037-1048, 2007.
- [7] А.П. Кудряшов. Извлечение и сопоставление точечных особенностей. Электронный научный журнал *Исследовано в России*, т.10, 2007. <http://zhurnal.ape.relarn.ru/articles/2007/104.pdf>

- [8] Y. Ke, R. Suthanakar. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. In *Computer Vision and Pattern Recognition*, volume 2, pages 506-513, 2004.
- [9] L.J. Heyer, S. Kruglyak and S. Yooseph. Exploring Expression Data: Identification and Analysis of Coexpressed Genes. In *Genome Research*, volume 9, pages 1106-1115, 1999.
- [10] A. Gionis, P. Indyk and R. Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the 25<sup>th</sup> VLDB Conference*, pages 518-529. 1999.
- [11] J.L. Bentley. K-d Trees for Semidynamic Point Sets. In *Proceedings of 6th Annual Symposium on Computational Geometry*, pages 187-197, 1990.
- [12] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. In *Journal of Applied Statistics*, volume 21(2), pages 224-270, 1994.
- [13] A.P. Witkin. Scale-space filtering. In *Proceeding of International Joint Conference on Artificial Intelligence*, pages 1019-1022, 1983.
- [14] A.B. Скурихин. Применение методов масштабируемого пространства в обработке сигналов.  
<http://www.spiiras.nw.ru/rus/conferences/ict/Skurihin110604.ppt>
- [15] J.J. Koenderink. The structure of images. In *Biological Cybernetics*, volume 50, pages 363-396, 1984.
- [16] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 128-142, 2002.
- [17] S.J. Cazzolato. Mapeo y localización simultánea en forma robusta basado en visión stereo. Ph.D. Thesis, 2007.
- [18] SIFT Keypoint Detector. <http://www.cs.ubc.ca/~lowe/keypoints/>
- [19] PCA-SIFT Source Code. <http://www.cs.cmu.edu/~yke/pcasift/>
- [20] Результаты для дорожки поиска нечетких дубликатов в коллекции изображений. Приложение к трудам РОМИП'2008.  
<http://romip.ru/>

# **Near-Duplicate Image Detection with Local Interest Point Extraction**

Vitaly Pimenov

Saint-Petersburg State University, Faculty of Applied Mathematics  
and Control Processes

vitaly.pimenov@gmail.com

The paper proposes a method for near-duplicate image detection by extraction, filtering and matching of local interest points with PCA-SIFT descriptors. For purposes of fast descriptor filtering an inverted index structure is proposed.

Experimental results on ROMIP'08 image collection show that proposed approach can be effectively applied for near-duplicate image detection tasks. The proposed inverted index structure is capable of increasing the matching performance up to 2 times.