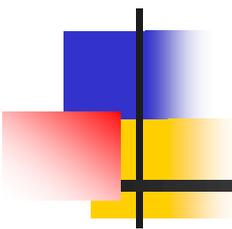
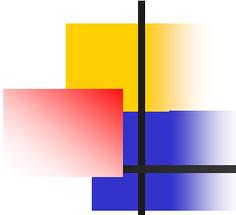


Обзор методов сопоставления шаблона и изображения



Алексей Федоров

Video Group
CS MSU Graphics & Media Lab



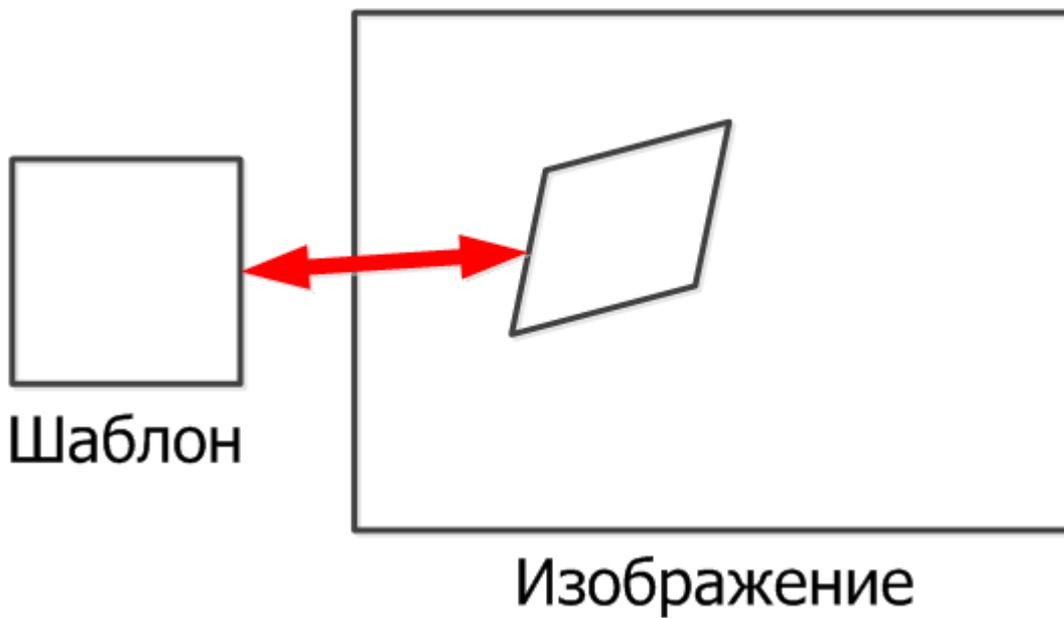
Содержание

- **Введение**
- Local Image Interest Point
- Template Matching Based on Fourier Coefficients
- Linear Scale and Rotation Invariant Matching
- Дополненная реальность
- Заключение

Введение

Постановка задачи

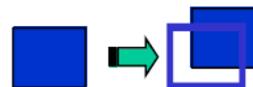
Устойчивый относительно геометрических искажений поиск шаблона на изображении



Введение

Геометрические преобразования

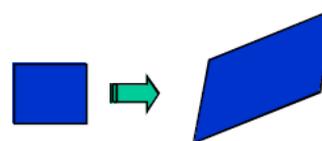
- Параллельный перенос



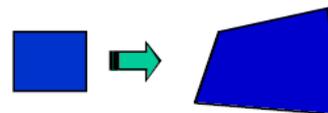
- Подобие (перенос, масштабирование, поворот)



- Аффинное



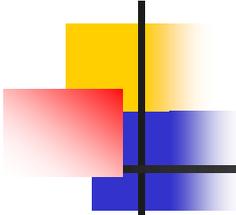
- Проективное



Введение

Простейший подход – «грубой силы»

1. Выбирается модель преобразования
2. Определяется набор параметров, описывающих модель
3. Выбирается функция сопоставления изображений
Sum of Absolute Differences (SAD), Sum of Squared Differences (SSD), Normalized Cross Correlation (NCC), etc.
4. Перебираются всевозможные значения параметров в разумных пределах



Содержание

- Введение
- **Local Image Interest Point**
- Template Matching Based on Fourier Coefficients
- Linear Scale and Rotation Invariant Matching
- Дополненная реальность
- Заключение

Local Image Interest Point

Схема работы

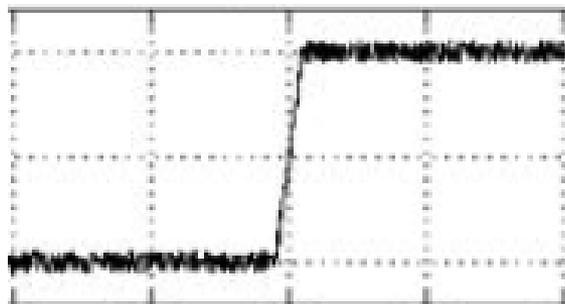


1. Выбираем ключевые точки
2. Применяем классическое вейвлет-преобразование к окрестностям ключевых точек и высокочастотные компоненты считаем особенностями
3. Сопоставляем ключевые точки

Название предложенного метода:
Rotation and Scale Invariant Matching (RASIM)

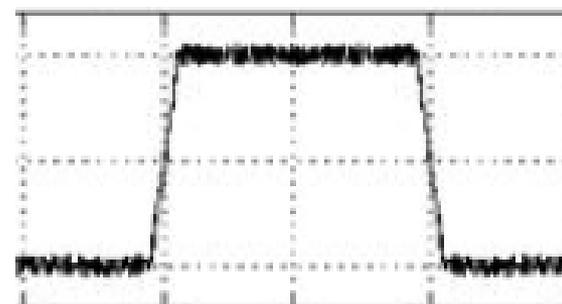
Блобы

Сигнал

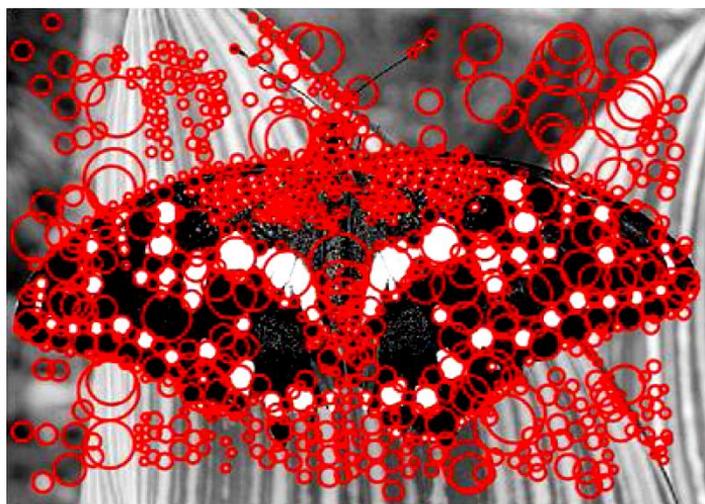


Край

Сигнал



Блоб

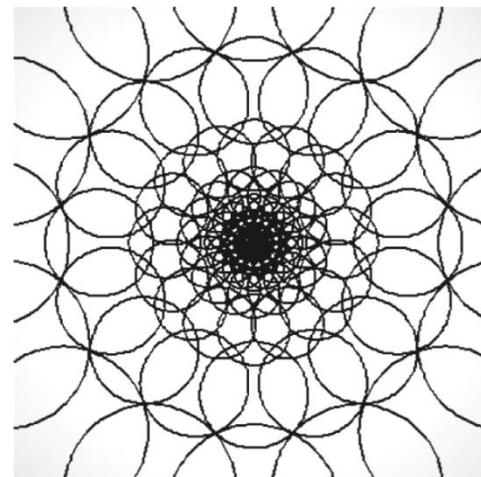


Пример блобов

Поиск блобов
осуществляется
с помощью разницы
Гауссиан (DoG)

Выбор ключевых точек

- Для поиска применяется псевдо-лог-полярная сетка дискретизации
- Ключевые точки находятся при поиске максимумов и минимумов в масштабированном изображении
- Масштабируется при помощи разницы Гауссиан (DoG)



Псевдо-лог-полярная
сетка дискретизации

Вейвлет-преобразование

$$\omega = (x * \tilde{h}^{\text{old}}) * \tilde{t}$$

$$\gamma^{\text{old}} = x * \tilde{g}^{\text{old}}$$

$$\gamma = \gamma^{\text{old}} - \omega$$

Быстрое вейвлет-преобразование

- x – входной сигнал
- $\tilde{g}^{\text{old}}, \tilde{h}^{\text{old}}$ – высокочастотный и низкочастотный фильтры
- γ, λ – высокочастотная и низкочастотная компоненты
- γ^{old} – старая высокочастотная компонента
- \tilde{t} – фильтр прогнозирования
- $*$ – операция свертки

Вейвлет-преобразование

Фильтр прогнозирования

p – размерность фильтра прогнозирования

v – количество выбранных больших высокочастотных компонент с индексами k'_1, k'_2, \dots, k'_v

$$\begin{bmatrix} \lambda_{k'_1} & \lambda_{k'_1-1} & \cdots & \lambda_{k'_1-p+1} \\ \lambda_{k'_2} & \lambda_{k'_2-1} & \cdots & \lambda_{k'_2-p+1} \\ \vdots & \vdots & \ddots & \cdots \\ \lambda_{k'_v} & \lambda_{k'_v-1} & \cdots & \lambda_{k'_v-p+1} \\ 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \tilde{t}_1 \\ \tilde{t}_2 \\ \vdots \\ \tilde{t}_p \end{bmatrix} = \begin{bmatrix} \gamma_{k'_1}^{\text{old}} \\ \gamma_{k'_2}^{\text{old}} \\ \vdots \\ \gamma_{k'_v}^{\text{old}} \\ 0 \end{bmatrix}$$

Фильтр прогнозирования получается решением системы для компонент, полученных от шаблонного изображения

Взвешенная адаптация

$\Lambda \mathbf{T} = \Gamma$ – система с предыдущего слайда

$$(\mathbf{w} \cdot \times \Lambda) \mathbf{T} = \mathbf{w} \cdot \times \Gamma$$

$$\mathbf{w} = [w_1, w_2, \dots, w_v, 1]^T$$

$$w_i = 1 + B \frac{(\gamma_{k'_i} - \gamma_{k'_v})}{(\gamma_{k'_1} - \gamma_{k'_v})}, \quad i = 1, \dots, v$$

$\gamma_{k'_1}, \gamma_{k'_v}$ – максимальная и минимальная компонента

B – некоторая константа

• \times – операция поэлементного умножения

Измерение подобия (1)

Метод разделен на две части: «оффлайн» и «онлайн»

«Оффлайн»:

1. Выбираем вейвлет преобразование и значения параметров p и $v \geq p - 1$
2. Вычисляем фильтр прогнозирования \tilde{t} .

Измерение подобия (2)

«Онлайн»:

1. Применяем вейвлет преобразование
2. Заполняем вектор D

$$D_k = \begin{cases} 0 & |\gamma_k| \geq |\gamma_k^{\text{old}}| \\ 100 - (100 |\gamma_k / \gamma_k^{\text{old}}|) & |\gamma_k| < |\gamma_k^{\text{old}}| \end{cases}, \forall k$$

3. Суммируем вектор в окне такой же длины ($2L$), как и сигнал шаблона

$$\tau_k = \sum_{l=k-L}^{k+L} D_l, \quad \forall k$$

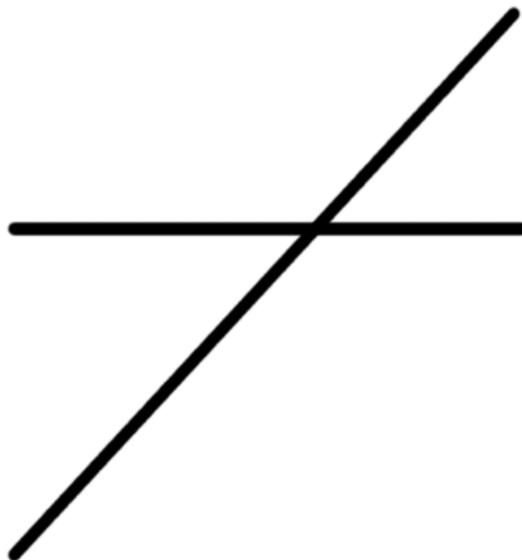
Максимальное значение суммы достигается при наилучшем соответствии

Постобработка

Преобразование Хафа

Метод предназначен для поиска объектов, принадлежащих определённому классу фигур с использованием процедуры голосования

Входное изображение

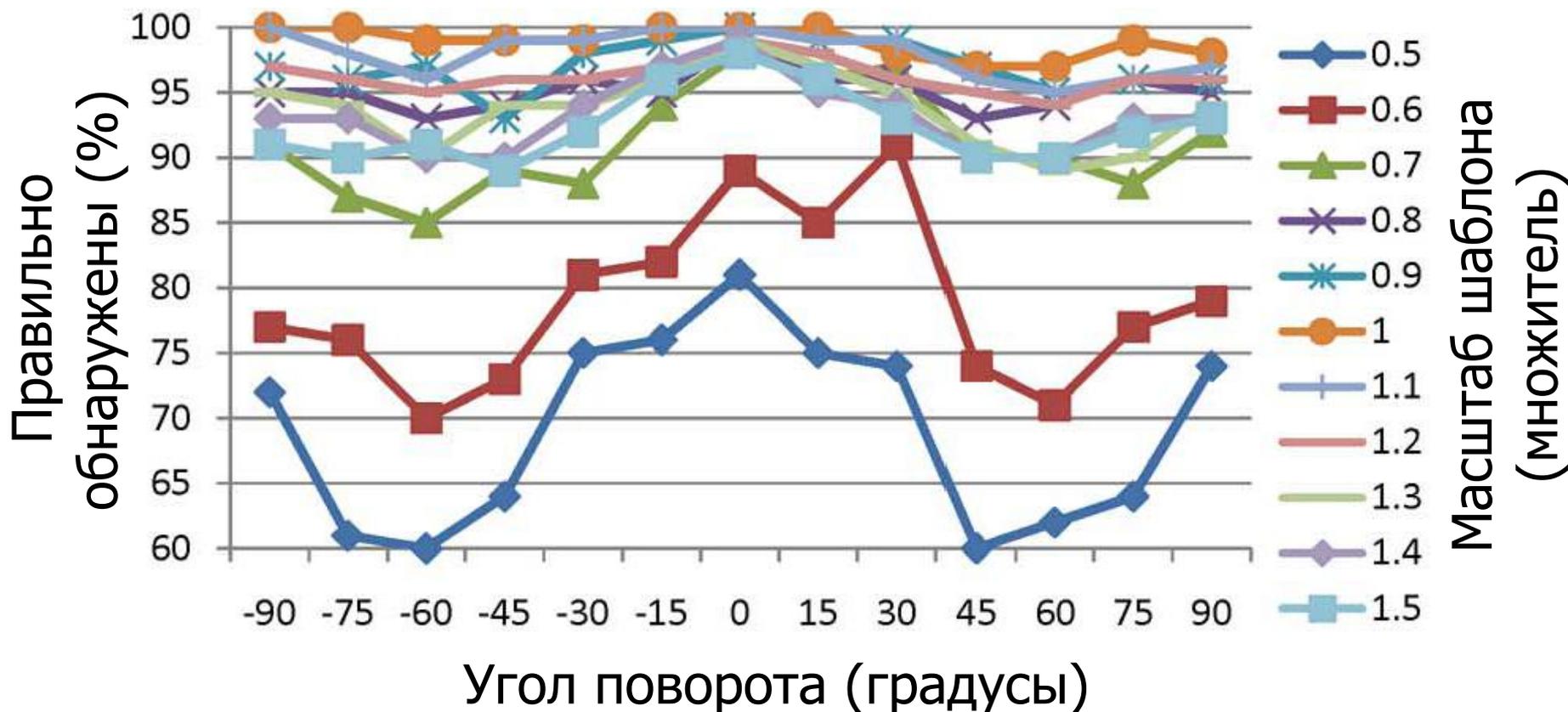


Преобразованное изображение



Результат

Статистика качества детектирования



Результат

Сравнение с конкурентом

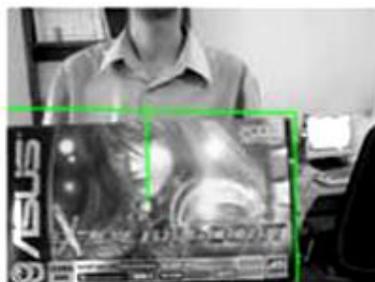
Reference Template



SIFT



✓



✓



✗



✗

RASIM



✓



✓



✓



✗

Frame#

1

138

571

671

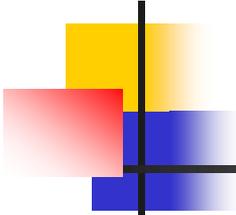
M. Amiri and H.R. Rabiee, "RASIM: A Novel Rotation and Scale Invariant Matching of Local Image Interest Points,"
IEEE Transactions on Image Processing, 2011

Статистика производительности



Test sequence	SIFT	SURF	RASIM
<i>boat</i>	48.1 (sec)	5.1 (sec)	53.6 (sec)
<i>graffiti</i>	45.2 (sec)	4.7 (sec)	49.3 (sec)

Время работы на одном изображении



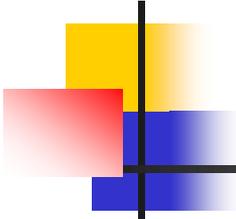
Выводы

Достоинство:

- Устойчивость к поворотам и масштабированию

Недостатки:

- Низкая скорость работы
- Неустойчивость к изменениям перспективы



Содержание

- Введение
- Local Image Interest Point
- **Template Matching Based on Fourier Coefficients**
- Linear Scale and Rotation Invariant Matching
- Дополненная реальность
- Заключение

Template Matching Based on Fourier Coefficients

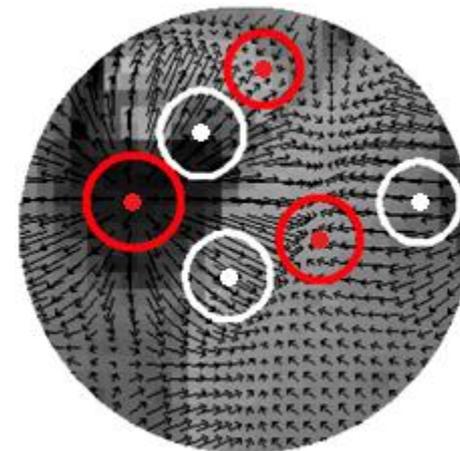
Обозначения:

Forapro – название предложенного алгоритма

A – исходное изображение

Q – шаблон

Стабильной называется точка, которая имеет некоторую окрестность, в которой направление градиента сохраняется

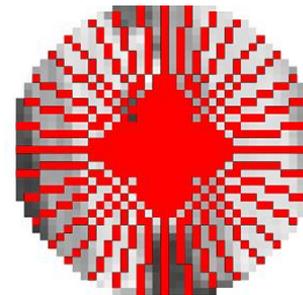


Белые – стабильные точки, красные – нет

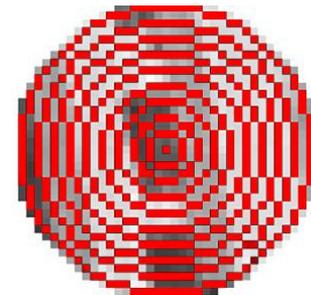
Forapro

Идеи метода (1)

- Значения цвета на радиальной и круговой проекциях в точке представляют собой значения дискретных функций
- Приблизим их непрерывными функциями и разложим последние в ряды Фурье
- Вычислим первые $L(K)$ комплексных коэффициентов $c_i(r_i)$ разложения круговой (радиальной) функции в ряд Фурье



Радиальная проекция в точке $C(A(x,y))$



Круговая проекция в точке $R(A(x,y))$

Forapro

Идеи метода (2)

Вычислим вектор особенностей v_f инвариантных к поворотам и масштабированию

$\angle r_k, |r_k|$ – полярный угол и длина комплексного числа

$$\text{dra}_k = \text{mod}(\angle r_k - k\angle r_1, 2\pi), \quad k \geq 2$$

$$v_{ra} = [\text{dra}_2, \text{dra}_3, \dots, \text{dra}_K] \quad v_{rm} = v[|r_1|, |r_2|, \dots, |r_K|]$$

$$v_{cf} = v[\text{re}(c_1), \text{im}(c_1), \text{re}(c_2), \text{im}(c_2), \dots, \text{re}(c_L), \text{im}(c_L)]$$

$$v_f = (v_{rm}, v_{ra}, v_{cf})$$

$\Lambda(v_f(A(x, y)), v_f(T(x_0, y_0)))$ – функция расстояния между двумя векторами особенностей

Forapro

Шаги алгоритма

1. Вычисляются значения коэффициентов c_{it} , r_i и v_f для всех точек A и Q
2. Выбираются N стабильных точек из Q , которые образуют множество $\{T\}$
3. Для i от 1 до N

Для каждого пикселя (x, y) из A вычисляются

$$D_i(x, y) = A(v_f(T_i(x_0, y_0)), v_f(A(x, y)))$$

Находится пиксель с самым малым отклонением и добавляется в массив

4. С помощью преобразования Хафа уточняется результат

Результаты (1)



Исходное изображение



Результат

H.Y. Kim, "Rotation-Discriminating Template Matching Based on Fourier Coefficients of Radial Projections with Robustness to Scaling and Partial Occlusion," Pattern Recognition, 2010

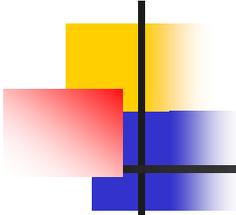
Результаты (2)



Исходное изображение



Результат



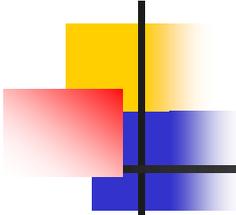
Выводы

Достоинства:

- Устойчивость:
 - к изменениям по яркости и контрасту
 - к поворотам и масштабированию

Недостатки:

- Невысокая скорость работы
- Неустойчивость к изменениям перспективы

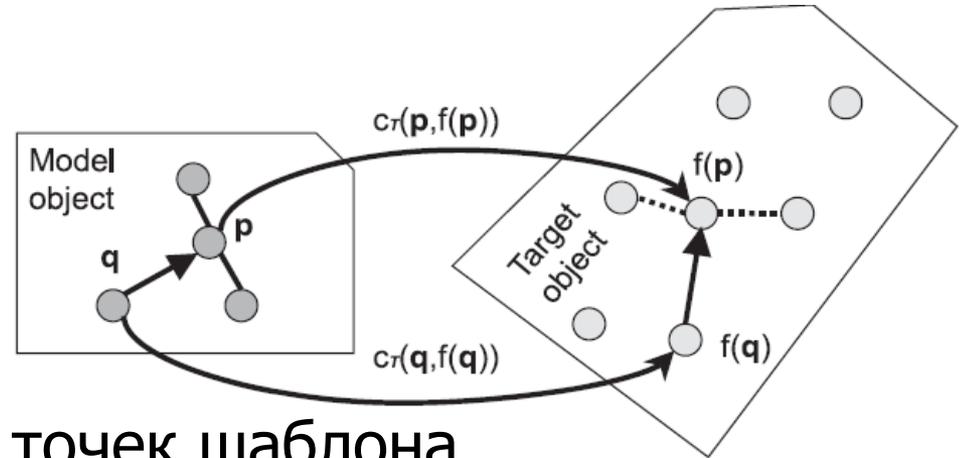


Содержание

- Введение
- Local Image Interest Point
- Template Matching Based on Fourier Coefficients
- **Linear Scale and Rotation Invariant Matching**
- Дополненная реальность
- Заключение

Linear Scale and Rotation Invariant Matching

Обозначения:



\mathcal{M} – множество точек шаблона

\mathcal{N} – множество пар точек из \mathcal{M}

p, q – некоторые точки из множества \mathcal{M}

$f(p), f(q)$ – точки, соответствующие p, q

$c_T(p, f(p))$ – стоимость сопоставления
(зависит от векторов особенностей)

Основная идея метода

Сопоставляем точки шаблона и изображения,
решая задачу линейного программирования (ЛП)

Стандартная постановка задачи ЛП:

Максимизировать функцию $F(x) = \sum_{j=1}^n c_j x_j$

при условиях $\sum_{j=1}^n a_{ij} x_j \leq b_i$ для $i = 1, 2, \dots, m$,

где c_j, a_{ij}, b_i – некоторые константы

Идеи метода

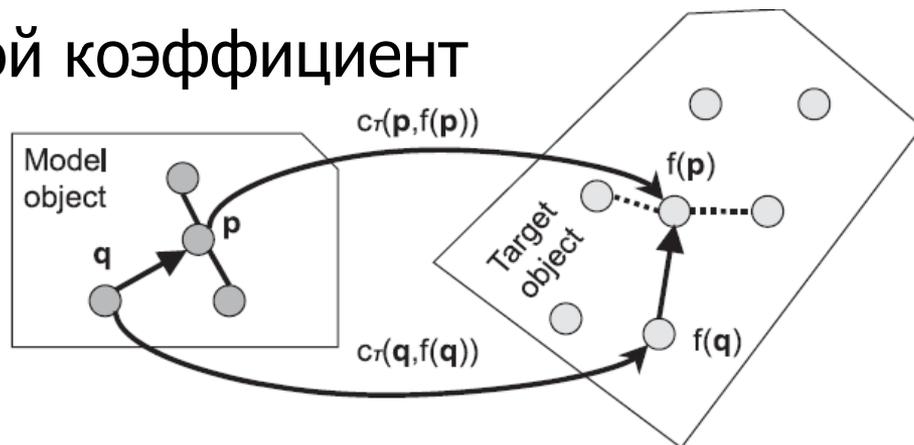
Критерий сопоставления (1)

$$\min_{f, T} \left\{ \sum_{p \in \mathcal{M}} c_T(p, f(p)) + \lambda \sum_{\{p, q\} \in \mathcal{N}} g_T(p, q, f(p), f(q)) \right\}$$

$c_T(p, f(p))$ – локальная стоимость сопоставления

$g_T(p, q, f(p), f(q))$ – пространственная стоимость сопоставления

λ – весовой коэффициент



Идеи метода

Критерий сопоставления (2)

$$\min_{\mathbf{f}, s, R} \left\{ \sum_{\mathbf{p} \in \mathcal{M}} c_{s, R}(\mathbf{p}, \mathbf{f}(\mathbf{p})) + \lambda \sum_{\{\mathbf{p}, \mathbf{q}\} \in \mathcal{N}} \|R \cdot (\mathbf{p} - \mathbf{q}) - s \cdot (\mathbf{f}(\mathbf{p}) - \mathbf{f}(\mathbf{q}))\| \right\}$$

$c_{s, R}(\mathbf{p}, \mathbf{f}(\mathbf{p}))$ – стоимость сопоставления,
инвариантная относительно
масштабирования и поворота

s – коэффициент масштабирования

R – матрица поворота (2×2)

Идеи метода

Критерий сопоставления (3)

$$\min \varepsilon(X, s, R) = \text{tr}(C'X) + \lambda |EMR - sEXT|$$

X – бинарная матрица сопоставления точек шаблона и изображения

C' – транспонированная матрица стоимостей сопоставления

E – матрица инцидентности ребер

M – координаты точки шаблона

T – координаты точки изображения

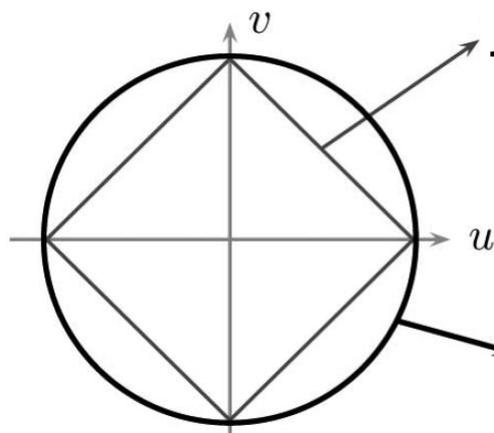
Идеи метода

Линеаризация

- Добавляются масштабированные шаблоны поиска

$$X = \sum_l X_l \quad X_l - \text{бинарная матрица сопоставления точек масштабированного шаблона и изображения}$$

- Изменяется матрица поворота $R = \begin{bmatrix} u & -v \\ v & u \end{bmatrix}$



Линеаризованные параметры:

$$u \pm v = \pm 1$$

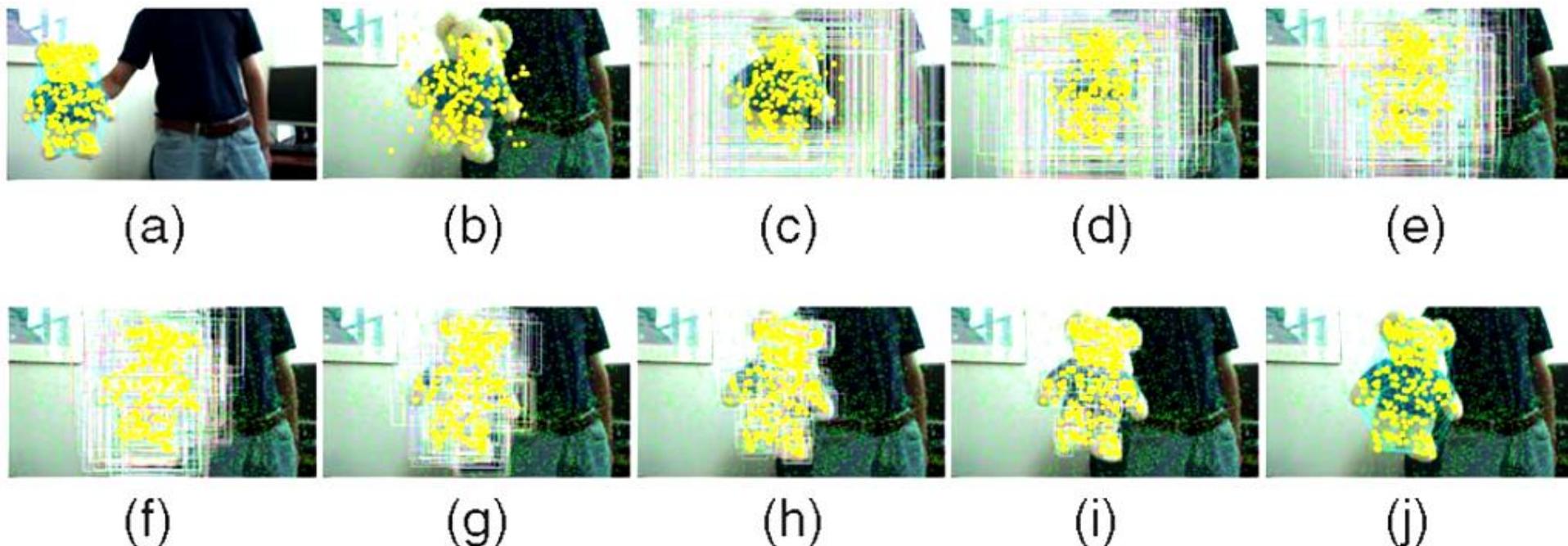
$$|u| \leq 1, |v| \leq 1$$

Исходные параметры: $u^2 + v^2 = 1$

Шаги алгоритма

1. Вычисление матрицы C стоимостей сопоставления точек шаблона и изображения
2. Выбор региона доверия для каждой точки шаблона
3. Вычисление минимальной выпуклой оболочки вершин из региона доверия для каждой точки шаблона
4. Решение задачи линейного программирования
5. Уменьшение регионов доверия. Если точки сопоставились хорошо, то выводятся результаты, иначе – перейти на шаг 3

Пример работы алгоритма



(a) – шаблон, (b) – начальное приближение,
 (c)–(i) – итерации алгоритма, (j) – результат
 Цветные прямоугольники – регионы доверия

Результат

Сравнение с конкурентами



O. Duchenne, et al.
"Tensor-Based
Algorithm for High-Order
Graph Matching," 2009

S. Belongie, et al.
"Shape Matching Using
Shape Contexts," 2002

Предложенный
алгоритм

Шаблон

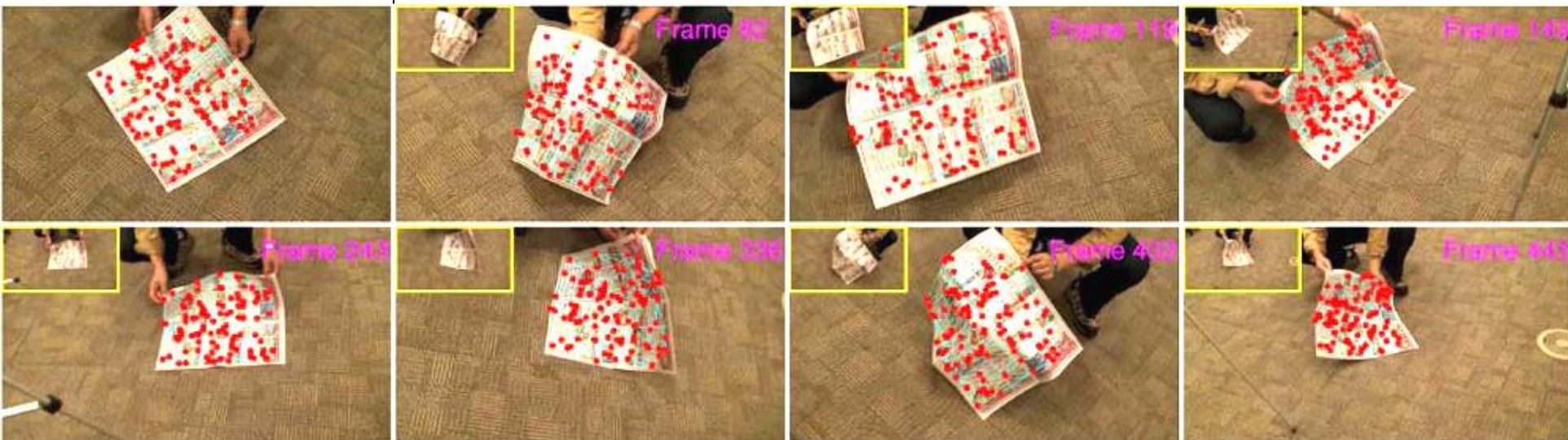
Кадры видеопоследовательности

Результат

Объекты с сильной деформацией (1)

Шаблон

Кадры видеопоследовательности



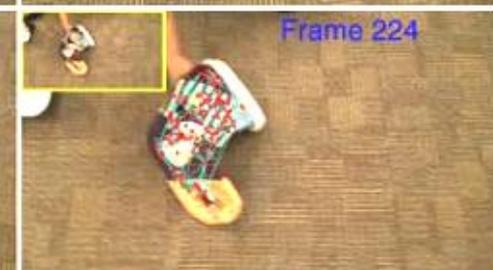
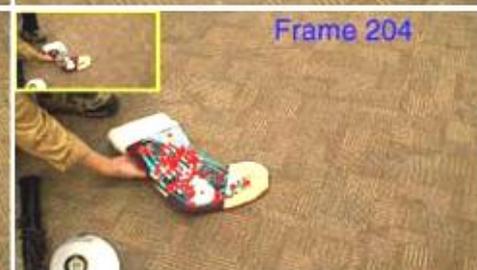
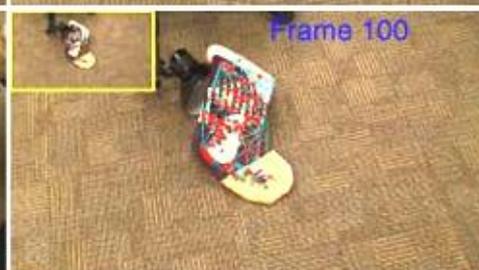
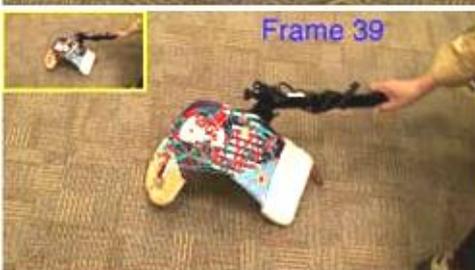
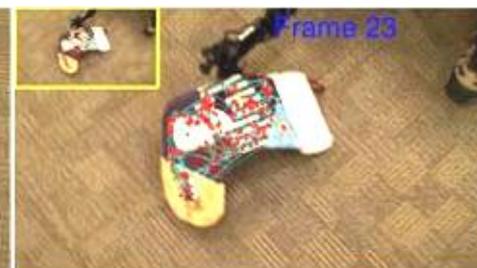
Кадры видеопоследовательности

Результат

Объекты с сильной деформацией (2)

Шаблон

Кадры видеопоследовательности



Кадры видеопоследовательности

Статистика производительности



video	<i>book</i>	<i>magazine</i>	<i>bear</i>	<i>butterfly</i>	<i>bee</i>	<i>fish</i>
#frames	856	601	601	771	101	131
#model	151	409	235	124	206	130
#target	2143	1724	1683	1405	1029	7316
time	1.6	11	2.2	1	2	0.9
accuracy	99%	97%	88%	95%	79%	95%

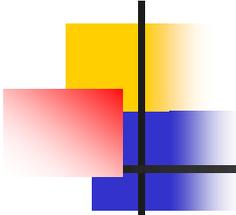
#frames – количество кадров в видеопоследовательности

#model – количество точек в шаблоне

#target – количество точек на изображении

time – время работы на 1 кадре (сек) на 2.8 GHz PC

accuracy – средняя точность по всей видеопоследовательности



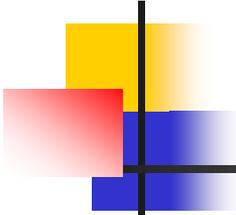
Выводы

Достоинства:

- Стабильность
- Высокая скорость работы

Недостаток:

- Не оговаривается, как выбирать точки для сопоставления



Содержание

- Введение
- Local Image Interest Point
- Template Matching Based on Fourier Coefficients
- Linear Scale and Rotation Invariant Matching
- **Дополненная реальность**
- Заключение

Пример применения

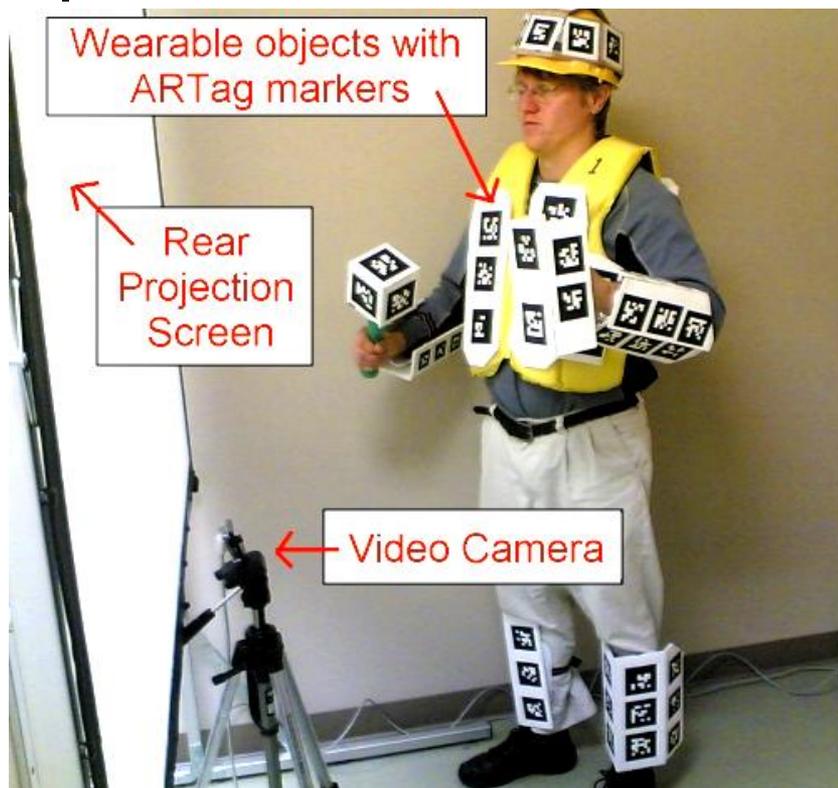
Дополненная реальность – добавление к поступающим из реального мира ощущениям мнимых объектов



Пример
дополненной
реальности

Дополненная реальность

Как это работает?



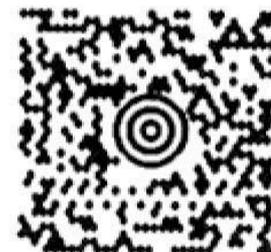
Находятся и распознаются специальные маркеры, затем дорисовывается нужная информация

Дополненная реальность

Системы маркеров



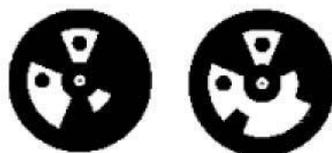
Data Matrix
(3 examples)



Maxicode
(US Postal Service)



QR (Quick Response Code)



Intersense



ARToolkit
(4 examples)



ARTag
(4 examples)



ARStudio



Некоторые системы маркеров

Дополненная реальность

Библиотеки

На данный момент существует несколько неплохих открытых реализаций поиска и трекинга маркеров:

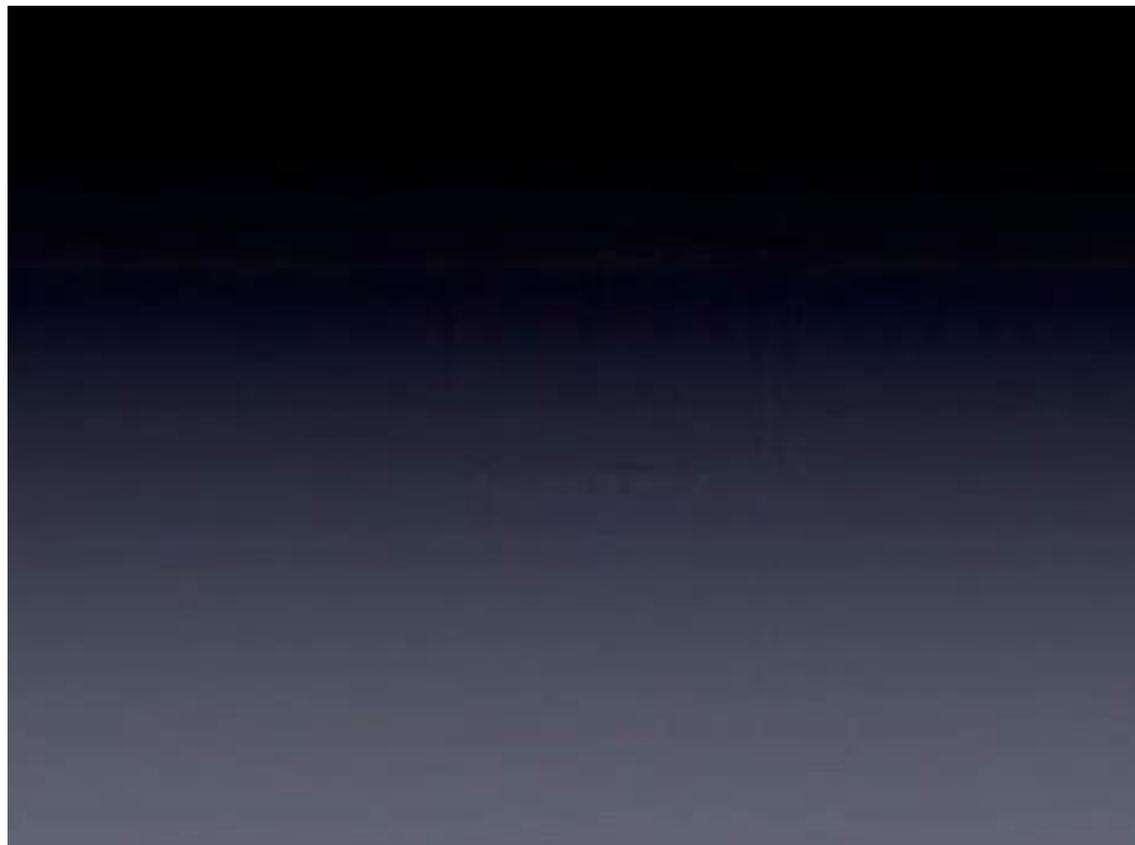
- ARToolKit (и дочерние)
- ZBar

...

В OpenCV по умолчанию трекинга маркеров нет

Дополненная реальность

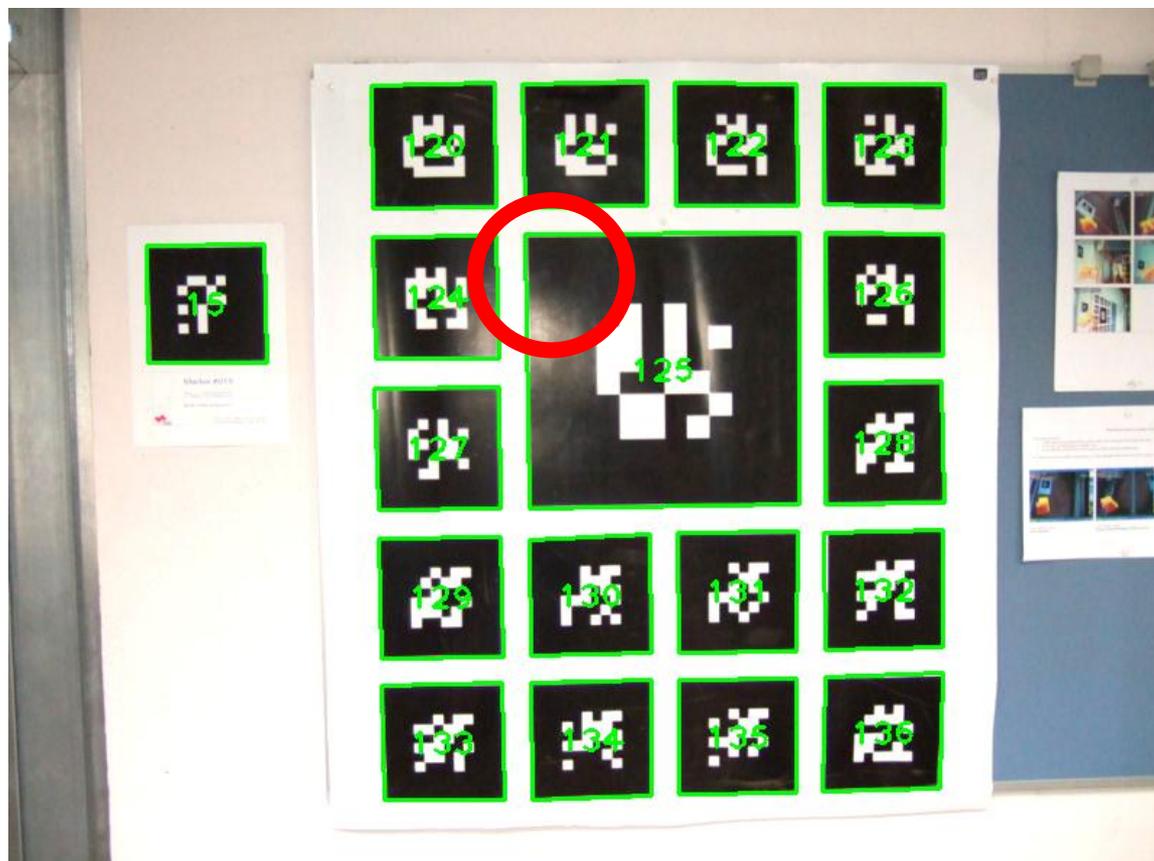
Поиск маркеров



Один из методов поиска маркеров

Результат

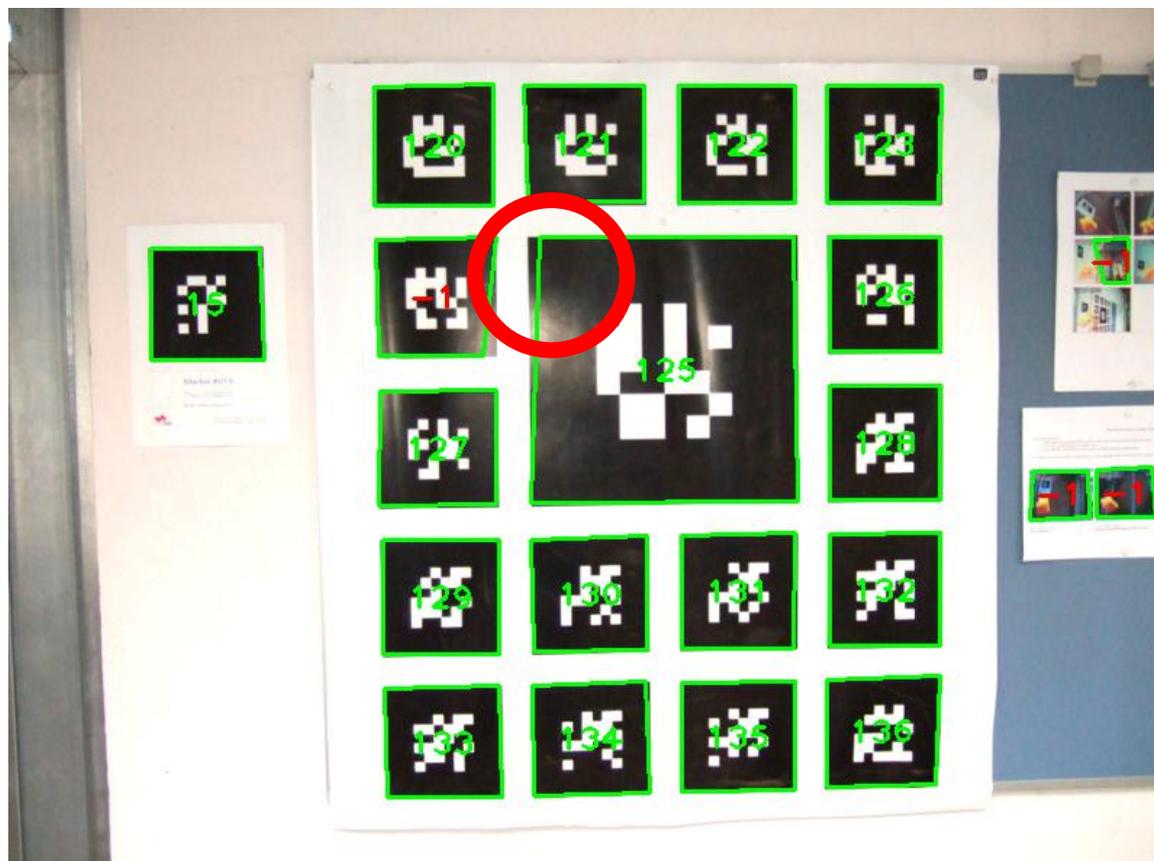
Сравнение с конкурентами (1)



Предложенный алгоритм

Результат

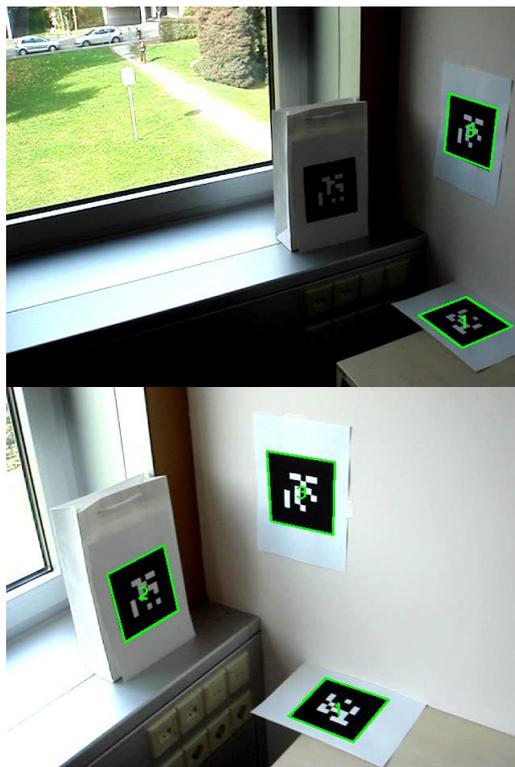
Сравнение с конкурентами (2)



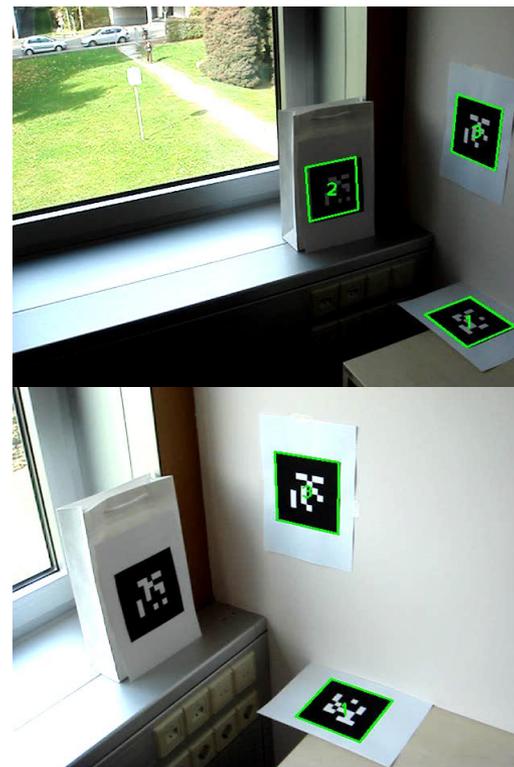
ARToolKit

Результат

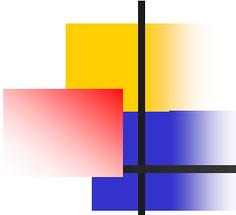
Сравнение с конкурентами (3)



Предложенный
алгоритм



ARToolKit



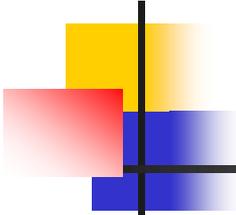
Выводы

Достоинства:

- Устойчивость к частичным перекрытиям
- Высокая скорость работы

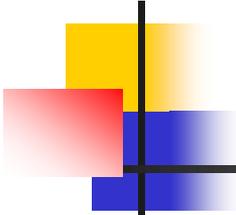
Недостаток:

- Неустойчивость к изменениям яркости
(не работает на темных объектах)



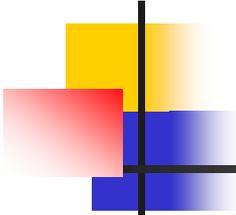
Содержание

- Введение
- Local Image Interest Point
- Template Matching Based on Fourier Coefficients
- Linear Scale and Rotation Invariant Matching
- Дополненная реальность
- **Заключение**



Заключение

- Рассмотренные алгоритмы с разных сторон подходят к решению задачи поиска и сопоставления шаблона и изображения
- У каждого есть свои плюсы и минусы, область применения



Проделанная работа

Протестирована работоспособность библиотеки ZBar по поиску маркеров

- Входные данные: изображения (19 мегапикселей), приближенные к «боевым»
- Время работы: 2–3 секунды
- Результат: все метки найдены

Проделанная работа

Результаты



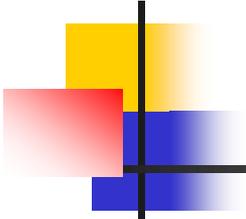
Изображение, приближенное к реальному тестированию устройств

Дальнейшие планы

- Создание программы-визарда для автоматического тестирования 3D-устройств
- Тестирование новых и уже имеющихся 3D-устройств
- Разработка системы тестирования стереоскопического зрения

Литература

1. M. Amiri and H.R. Rabiee, "RASIM: A Novel Rotation and Scale Invariant Matching of Local Image Interest Points," *IEEE Transactions on Image Processing*, vol. 20, 2011, pp. 3580–3591.
2. H.Y. Kim, "Rotation-discriminating Template Matching Based on Fourier Coefficients of Radial Projections with Robustness to Scaling and Partial Occlusion," *Elsevier Pattern Recognition*, vol. 43, 2010, pp. 859–872.
3. H. Jiang, S.X. Yu, and D.R. Martin, "Linear Scale and Rotation Invariant Matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, 2011, pp. 1339–1355.
4. Martin Hirzer, "Marker Detection for Augmented Reality Applications," *Project Image Analysis Graz*, 2008, pp. 1–25.
5. А. Конушин, «Сопоставление изображений и локальные особенности», лекции по курсу «Введение в компьютерное зрение», 2012, стр. 45–79.



Вопросы?
Предложения?
Наряды?

Лаборатория компьютерной графики и мультимедиа



Видеогруппа — это:

- Выпускники в аспирантурах Англии, Франции, Швейцарии (в России в МГУ и ИПМ им. Келдыша)
- Выпускниками защищены 5 диссертаций
- Наиболее популярные в мире сравнения видеокодеков
- Более 3 миллионов скачанных фильтров обработки видео