

2. Полежаев П.Н. Исследование алгоритмов планирования параллельных задач для кластерных вычислительных систем с помощью симулятора // Параллельные вычислительные технологии (ПАВТ–2010): тр. междунар. конф. – Челябинск: Изд-во ЮУрГУ. 2010. С. 287 – 298.

С.В. Роцин, И.И. Зиновьев

Владимирский государственный университет им. А.Г. и Н.Г. Столетовых

ЗАДАЧА ОБНАРУЖЕНИЯ ЛИЦ НА ИЗОБРАЖЕНИИ С ИСПОЛЬЗОВАНИЕМ СОВРЕМЕННЫХ КОМПЬЮТЕРОВ

Задача обнаружения лица на изображении (face detection) часто является «первым шагом», предобработкой в процессе решения задачи «более высокого уровня» (например, узнавание лица, распознавание выражения лица). Однако и сама информация о присутствии и, возможно, количестве лиц на изображении или в видеопотоке может быть полезна для таких приложений, как охранные системы и содержательная индексация базы данных изображений или видеофрагментов.

Существующие алгоритмы обнаружения лиц можно разбить на две широкие категории. К первой категории относятся методы, отталкивающиеся от опыта человека в распознавании лиц и делающие попытку формализовать и алгоритмизовать этот опыт, построив на его основе автоматическую систему распознавания. Вторая категория опирается на инструментарий распознавания образов, рассматривая задачу обнаружения лица как частный случай задачи распознавания.

Основа методов первой категории – эмпирика является одновременно их сильной и слабой стороной. Трудно эффективно перевести неформальный человеческий опыт и знания в набор формальных правил, поскольку чересчур жесткие рамки правил приведут к тому, что в ряде случаев лица не будут обнаружены и, напротив, слишком общие правила приведут к большому количеству случаев ложного обнаружения. Методы вто-

рой категории дальше продвинулись в качестве обнаружения лиц на изображении. Но достигнуто это ценой их высокой вычислительной сложности. Во-первых, сами классификаторы часто включают в себя большое количество достаточно сложных вычислений; во-вторых, полный перебор всех возможных прямоугольных фрагментов изображения сам по себе занимает большое количество времени. Это затрудняет использование некоторых методов в системах реального времени (например, отслеживание перемещения лица в видеопотоке).

В качестве примера алгоритма второй категории возьмем классификатор Viola-Jones [1], в основе которого лежит алгоритм boosting (усиления слабых классификаторов). Для тестирования взята реализация данного алгоритма в библиотеке OpenCV (англ. *Open Source Computer Vision Library*, библиотека компьютерного зрения с открытым исходным кодом) [3]. На практике алгоритм показал высокий уровень обнаружений и низкий уровень ложных срабатываний (рис. 1).

В OpenCV представлен однопоточный вариант алгоритма, что не позволяет использовать все возможности современных компьютеров. Поэтому был реализован собственный многопоточный вариант алгоритма. Поскольку задача поиска лица на изображении чаще всего локализована на одном компьютере и применение вычислительного кластера невозможно, для реализации многопоточного алгоритма использовалась технология OpenMP. Анализ алгоритма работы классификатора Viola-Jones и исходного кода библиотеки OpenCV показал, что сам процесс классификации участка изображения занимает немного времени, основную вычислительную сложность представляет перебор всех возможных прямоугольных участков изображения. Именно эта часть алгоритма и была распределена между всеми доступными процессорами. Для хранения промежуточных результатов используется разделяемый между потоками массив. В конце работы алгоритма главный поток обрабатывает полученные промежуточные данные с целью усреднения результатов, так как в результате работы алгоритма одно лицо может обнаруживаться несколько раз в разных масштабах.



Рис. 1. Результат работы классификатора Viola-Jones

Временные характеристики работы многопоточного алгоритма Viola-Jones приведены на рис. 2.

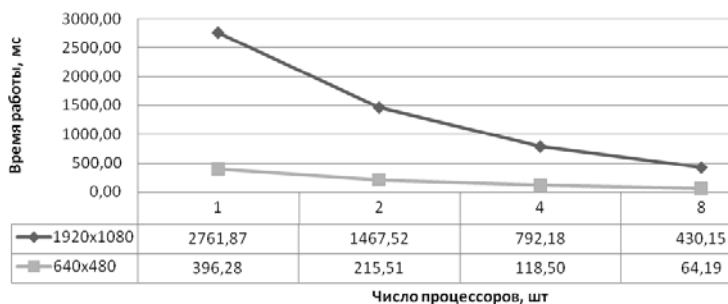


Рис. 2. Временные характеристики работы классификатора Viola-Jones

Из проведенных экспериментов становится ясно, что современные персональные компьютеры вполне справляются с задачей поиска лиц в видеопотоке с обычной веб-камеры в реальном времени. Для обработки видео высокого разрешения необходимы более эффективные алгоритмы.

Список литературы

1. Paul Viola, Michael J. Jones. Robust real-time face detection // International journal of computer vision. – 2004, 57(2). – P. 137–154.

2. Cha Zhang and Zhengyou Zhang. A Survey of Recent Advances in Face Detection / Microsoft Research Technical Report, 2010.
3. <http://opencv.willowgarage.com/wiki>.
4. <http://openmp.org/wp>.

А.В. Русаков

Нижегородский государственный университет им. Н.И. Лобачевского

ОБ ОДНОМ ПОДХОДЕ К СРАВНИТЕЛЬНОМУ АНАЛИЗУ ТЕХНОЛОГИЙ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ В СИСТЕМАХ С ОБЩЕЙ ПАМЯТЬЮ

В настоящее время наблюдается неуклонный рост производительности вычислительной техники, что открывает все новые возможности для решения ресурсоемких задач, имеющих большое практическое значение. Такие задачи можно встретить в физике, химии, биологии, экономике, финансах и т.д. При этом одно из основных направлений развития состоит в частичном дублировании устройств, интегрированных на кристалле центрального процессора. Наличие и распространение таких вычислительных систем потребовало новых технологий, моделей, методов и программных средств системного уровня, ориентированных на их эффективное использование. Так, в научной периодике наблюдается неуклонный рост числа публикаций, связанных с новыми технологиями для параллельного программирования. Разрабатываются новые подходы (OpenMP, MPI), фреймворки (OpenCL, CUDA), библиотеки (TBB, CCR), языки (Cilk, Erlang, APL, Sisal) параллельного программирования. Одна из основных проблем заключается в сравнительной сложности разработки параллельных программ с нуля. Переход от обычного последовательного кода к параллельному требует определенных навыков системного программирования, не всегда присутствующих у прикладных программистов – ключевых потребителей новых технологий параллельных вычислений. Таким образом, на первый план выходят два вопроса: производительность получаемых параллельных программ и трудоемкость их разработки.