Оценка производительности корреляционных мер сходства в задачах полного поиска движения на процессоре Л1879ВМ1(NM6403)

Мушкаев С.В.
Москва, НТЦ "Модуль", e-mail: mushkaev@module.ru

Аннотация

В задачах корреляционного анализа, связанных с поиском похожих объектов, поиском по маске, распознаванием и др. очень часто приходится обрабатывать довольно большие объемы данных, что зачастую приводит к большим вычислительным затратам. Как правило, для сравнения блоков данных используются две меры сходства: среднеквадратичную и среднеабсолютную ошибку. В данной статье изучается вопрос применимости этих корреляционных мер к процессору NM6403. Показывается высокая эффективность процессора при вычислении среднеабсолютных мер. А также подчеркивается тот факт, что при определенных условиях, на процессоре NM6403 среднеквадратичная мера может в несколько раз превосходить по скорости среднеабсолютную оценку.

Введение

В стандартах сжатия видео: MPEG-1, MPEG-2, H.261, H.263 и др. широко применяются блочные методы поиска и компенсации движения. В качестве меры сходства между двумя квадратными сегментами текущего изображения – C и предыдущего – R , как правило, используются два критерия: MSE-(Mean Square Error) и MAE(Mean Absolute Error). Среднеквадратичная (MSE) и среднеабсолютная (MAE) меры сходства двух блоков размером 16x16 пикселей на изображениях R и C и смещенных друг относительно друга на вектор (vx, vy) имеют вид:

$$MSE(x,y,vx,vy) = \frac{1}{256}\sum_{dx=0}^{15}\sum_{dy=0}^{15}(C_{x+dx,y+dy} - R_{x+vx+dx,y+vy+dy})^2 \tag{1}$$

$$MAE(x,y,vx,vy) = \frac{1}{256}\sum_{dx=0}^{15}\sum_{dy=0}^{15}|C_{x+dx,y+dy} - R_{x+vx+dx,y+vy+dy}| \tag{2}$$

На практике, нормализующий множитель $\frac{1}{256}$ не используется, и вместо мер MSE и MAE применяются эквивалентные меры SSD (Sum of Sqaured Differences) и SAD (Sum of Absolute Differences) соответственно:

$$SSD(x,y,vx,vy) = \sum_{dx=0}^{15}\sum_{dy=0}^{15}(C_{x+dx,y+dy} - R_{x+vx+dx,y+vy+dy})^2 \tag{3}$$

$$SAD(x,y,vx,vy) = \sum_{dx=0}^{15}\sum_{dy=0}^{15}|C_{x+dx,y+dy} - R_{x+vx+dx,y+vy+dy}| \tag{4}$$

Среднеквадратичная мера является более точной, однако среднеабсолютная мера на практике более предпочтительна с точки зрения экономии вычислительной мощности. Так разница в быстродействии этих способов на процессоре Intel Pentium может достигать порядка 40%

Однако, в ряде практических задач, таких как, поиск наилучшего соответствия, поиск движения и прочих приложениях, где производится многократное сравнение различных объектов с заданным шаблоном, ситуация может оказаться обратной - мера SSD может превосходить по производительности меру SAD. Определяющим фактором в таких случаях будет являться алгоритм поиска и архитектура процессора. Ниже будет рассмотрен подход именно к такого рода задачам применительно к архитектуре процессора NM6403. Для сравнения приводятся оба варианта реализаций мер SAD и SSD на процессоре NM6403.

1 Вычисление среднеабсолютной меры на процессоре NM6403

Вычисление меры SAD по формуле (4) на процессоре NM6403 наиболее быстрым способом осуществляется в три этапа:

1. Поэлементная разность двух матриц размером 16x16 пикселей
2. Вычисление 256 модулей быстрым способом
3. Суммирование 256 модулей

За счет параллельной обработки данных на векторном процессоре при 8-разрядных входных данных и 32-битном результате суммы каждый этап требует порядка 32 процессорных тактов. Суммарное вычисление меры SAD составляет около 96 тактов. Следует также учесть, что поскольку разность вычисляется в 8-разрядной сетке, то для избежания переполнения все значения пикселей должны лежать в диапазоне от 0 до 127.

2 Вычисление среднеквадратичной меры на процессоре NM6403

Раскроем квадрат в выражении (3):

$$\begin{aligned} &SSD(x,y,vx,vy) = \\ &= \textstyle\sum_{dx=0}^{15}\sum_{dy=0}^{15} C^2_{x+dx,y+dy} + \\ &+ \textstyle\sum_{dx=0}^{15}\sum_{dy=0}^{15} R^2_{x+vx+dx,y+vy+dy} - \\ &-2\textstyle\sum_{dx=0}^{15}\sum_{dy=0}^{15} C_{x+dx,y+dy}R_{x+vx+dx,y+vy+dy} \end{aligned} \tag{5}$$

Как можно видеть, в первое и второе слагаемое входят значения пикселей только одного изображения. Эти суммы можно вычислить заранее для всех x, y кадров C и R, получив матрицы [Sc] и [Sr] соответсвенно:

$$Sc_{x,y} = \sum_{dx=0}^{15}\sum_{dy=0}^{15} C^2_{x+dx,y+dy}$$

$$Sr_{x,y} = \sum_{dx=0}^{15}\sum_{dy=0}^{15} R^2_{x+dx,y+dy}$$

В этом случае выражение (5) сводится к более простому:

$$SSD(x,y,vx,vy) = Sc_{x,y} + Sr_{x+vx,y+vy} - 2\sum_{dx=0}^{15}\sum_{dy=0}^{15} C_{(x+dx,y+dy)}R_{(x+vx+dx,y+vy+dy)} \tag{6}$$

Последнее слагаемое есть суть операции взвешенного умножения с накоплением, аппаратно поддерживаемой процессором NM6403.

В формулу (6) входит две операции суммирования $Sc_{x,y}$ и $Sr_{x+vx,y+vy}$, 256-ть умножений с накоплением и одно умножение на -2. При разрядности входных данных – 8 бит, и разрядности выходных данных – 32 бита, матричный умножитель позволяет за один процессорный такт выполнить 16 умножений с накоплением [1]. Таким образом, время вычисления формулы (6) составляет 3+256/16=19 тактов. Дополнительной оптимизации можно достичь за счет совмещения одной операций суммирования и умножения на -2 с другими арифметическими действиями. Время вычисления формулы (6) в этом случае может быть снижено до 17 тактов.

3 Особенности применения

Приведенные выше цифры производительности мер SAD и SSD показывают принципиальную возможность и эффективность их использования на процессоре NM6403. Однако, следует отметить условия применимости данных методов. Указанные цифры являются ориентировочными, так как они справедливы только при достаточно большой зоне поиска, т.е когда каждый эталон сравнивается с большим числом кандидатов. Это связано со следующими двумя причинами:

1. В приведенном расчете не учитывались потери, связанные с перегрузкой матрицы весовых коэффициентов. При достаточно большом объеме сравниваемых данных временем первоначальной загрузки весовых коэффициентов (32 такта) можно пренебречь, остальные же перегрузки производятся на фоне вычислений.

2. В расчете также не учитывалось время предварительного вычисления матриц [Sc] и [Sr] и подготовки других дополнительных данных. Полное время обработки состоит из этой подготовительной части и собственно корреляционно-вычислительной части. Подготовительная часть не зависит от зоны поиска и чем больше будет размер зоны поиска тем менее существенной будет ее время по сравнению с корреляционной частью. При зонах поиска от 32x32 и более подготовительная часть составляет менее 10% всего времени и ей можно пренебречь.

4 Выводы

Приведенные методы демонстрируют высокую эффективность процессора при вычислении блочных среднеквадратичных (SSD) и среднеабсолютных (SAD) мер сходства. Так при больших зонах поиска время вычисления меры SAD для блоков размером 16x16 пикселей составляет порядка 100 тактов, а время вычисления SSD меры 17-25 тактов. Т.е. более точная среднеквадратичная мера сходства оказывается на процессоре NM6403 и в несколько раз менее трудоемкой, чем среднеабсолютная. При этом, основной проблемой в таком подходе вычисления SSD меры является достаточно трудоемкая процедура вычисления квадратов. Для решения этой проблемы можно применять как табличные методы, так и быстрые приближенные способы вычисления квадратов. Таким образом, процессор NM6403 становится особенно эффективным в задачах оценки движения методом полного перебора с большой зоной поиска (32x32 и более).

Список литературы

[1] НТЦ "Модуль" "Архитектура процессора Л1879ВМ1 (NM6403)" http://www.module.ru/files/nm6403arch-r.pdf