Создание системы распределенного отказоустойчивого хранения цветных крупноформатных изображений

Попов С. Б.
spop@smr.ru
Самара, Институт систем обработки изображений РАН

В настоящий момент растет интерес к системам параллельной или распределенной обработки изображений. В первую очередь это связано с тем, что появилась насущная потребность в обработке крупноформатных изображений. Наблюдается устойчивая тенденция к увеличению размеров формируемых изображений во многих областях деятельности. Однако увеличение размера изображения порождает большие проблемы при их обработке, хранении и передаче данных.

Обработка изображений с использованием параллельных или распределенных систем помогает преодолеть «проклятие размерности» в процессе вычислений, но для большинства таких систем обработки изображений узким местом являются предварительный этап рассылки данных исходных изображений по компьютерам распределенной вычислительной системы и завершающий этап сбора обработанных фрагментов в единое изображение. Попытки распараллелить или существенно сократить эти необходимые, но непроизводительные этапы распределенной обработки изображений приводят к идее распределенного изображения.

Распределенное изображение — это структура данных, определяющая способ и параметры разбиения изображения на фрагменты, список компьютеров, где находятся эти фрагменты, место их размещения и формат хранения. В данном случае фрагменты обрабатываемых изображений хранятся непосредственно на компьютерах, выполняющих параллельную обработку. Каждая задача параллельной программы обрабатывает тот фрагмент изображения, который расположен на компьютере, где выполняется данная задача, результат обработки сохраняется здесь же, как часть нового распределенного изображения, полученного в результате работы всех задач, участвующих в распределенной обработке.

Несмотря на очевидность данной идеи, она не получила распространения потому, что исследователи не смогли предложить удовлетворительных решений возникающих при этом проблем.

— Как выполнить декомпозицию (разбиение), близкую к оптимальной, для априори неизвестной последующей задачи обработки?
— Как решить проблемы сбалансированности загрузки компьютеров, участвующих в обработке при заранее выполненной декомпозиции данных?

— Как обеспечить достаточный уровень отказоустойчивости распределенного хранения фрагментов изображений?
— Как обеспечить виртуальную целостность распределенного изображения?
— Как обеспечить приемлемый уровень интерактивности системы при визуализации распределенных изображений?

В данной работе предлагается новый подход к организации хранения данных в виде распределенных изображений при параллельной обработке на многопроцессорных системах различной архитектуры.

Анализируя варианты декомпозиции изображений при параллельном выполнении различных операций обработки, следует отметить, что наиболее целесообразным для распределенного изображения представляется декомпозиция в виде перекрывающихся фрагментов. Особенность предлагаемого подхода заключается в том, что размер необходимого перекрытия фрагментов определяется не параметрами последующей задачи обработки, поскольку она априори неизвестна, а необходимостью решения проблем сбалансированности загрузки компьютеров и обеспечения достаточного уровня отказоустойчивости распределенного хранения фрагментов изображений.

Распределенное изображение определяется в виде набора *перекрывающихся* фрагментов изображения. Для каждого из M компьютеров фрагмент формируется следующим образом. Все строки изображения делятся на $2M$ блоков одинакового размера. Фрагмент распределенного изображения на m-м компьютере содержит два основных блока с номерами $2m-1$ и $2m$, а также два так называемых теневых блока с номерами $2m-2$ и $2m+1$, которые являются основными для двух соседних узлов, соответственно для $(m-1)$-го и $(m+1)$-го. Два основных блока соответствуют варианту декомпозиции изображения на непересекающиеся фрагменты, причем младший основной блок хранится в качестве одного из теневых блоков на компьютере с меньшим номером, а старший — на компьютере с большим номером. Таким образом, изображение разбивается на непересекающиеся фрагменты, а затем к фрагменту добавляются с каждой стороны по половине примыкающего фрагмента, хранящегося на соседнем компьютере. В распределенном изображении за каждым узлом хранения закреплен определенный фрагмент изображения (основные данные), части соседних фрагментов (теневые данные) хранятся здесь дополнительно. Именно данные основных блоков формируются на узле в процессе распределенной обработки изображения. По окончании процесса обработки выполняется обмен теневыми данными.

Такое заведомо избыточное разбиение решает проблему отказоустойчивости, то есть позволяет восстановить изображение при отказе одного из узлов хранения.

Одновременно с этим, предложенный принцип декомпозиции распределенного изображения позволяет реализовать оригинальный алгоритм динамического распределения нагрузки при выполнении операций поэлементной обработки или локальной обработки скользящим окном. Суть его заключается в следующем. Каждый вычислительный узел, участвующий в параллельной обработке, начинает вычисления с первой строки старшего основного блока, затем формируется последняя строка младшего блока. Далее попеременно формируются строки старшего блока в порядке возрастания номера и строки младшего блока в порядке убывания. Когда общее количество сформированных на узле строк достигнет некоторого определенного значения, например половины количества строк в блоке основных данных, узлам, обрабатывающим соседние фрагменты, рассылаются сообщения о том, что текущий узел на четверть завершил свою работу. Если к этому времени соответствующих сообщений от соседей не получено, то это означает, что текущий компьютер должен запланировать себе формирование и тех строк результирующего изображения, которые являются для него теневыми.

Рассматривая взаимодействие двух соседних вычислительных узлов, можно заметить, что они совместно формируют ту часть результирующего изображения, которая содержится между серединами их основных данных, причем каждый из них имеет всю необходимую информацию, чтобы сделать эту работу самостоятельно. В процессе работы смежные узлы двигаются навстречу друг другу, сообщая о скорости своего процесса вычислений при достижении заранее определенных моментов, например четверти всей работы, половины, трех четвертей и, наконец, полного завершения. При этом прогнозируемый номер строки изображения, на которой эти процессы встретятся, постоянно корректируется в зависимости от текущей загрузки вычислительных узлов. Таким образом, все процессы завершат свою работу практически одновременно. Разница при этом составит на больше, чем время обработки одной строки. В результате будет сформировано результирующее изображение в полном объеме, но размещение его данных по компьютерам, содержащим новое распределенное изображение, будет неравномерным. Однако пользователь может получить результат своего запроса сразу по завершении процесса обработки. Далее в фоновом режиме узлы хранения только что сформированного распределенного изображения обменяются своими данными для того, чтобы привести структуру распределенного изображения к необходимому виду.

При реализации функции визуализации крупноформатных распределенных изображений приемлемый уровень интерактивности системы обеспечивает наличие так называемого эскиза изображения на каждом или некоторых узлах распределенной системы хранения. В этом случае визуализирующее приложение, обращаясь к узлу с наиболее быстрым доступом, получает данные, необходимые для предварительного просмотра изображения. Количество узлов хранения, которые содержат эскизы, может выбираться исходя из конфигурации локальной сети рабочей группы пользователей, совместно использующих распределенную систему хранения, или допустимого уровня избыточной информации, которая может храниться в такой системе. При просмотре изображения в полноразмерном режиме пользователю отображается только выбранный им фрагмент. При прокрутке изображения в окне происходит подкачка необходимых данных. Для обеспечения комфортного уровня интерактивности подкачка выполняется в режиме чтения с упреждением (prepaging). Наличие перекрытий хранящихся фрагментов делает этот процесс более плавным.

Таким образом, предложенный подход к организации данных в распределенных изображениях снимает большинство из обозначенных выше проблем.

Работа выполнена при поддержке РФФИ, проект № 07-07-00210.