**РАЗРАБОТКА МЕТОДА АНАЛИЗА ВИДЕО ПОТОКА**

Зорин С.А.

Научный руководитель – к.т.н., доц. Машталир С.В.

Харьковский национальный университет радиоэлектроники

(61166, Харьков, пр. Ленина, 14, каф. Информатики, тел. (057)702-14-19),

E-mail: [sergeyzorin89@gmail.com](mailto:sergeyzorin89@gmail.com);

This work is directed to the modern developments and researches in the field of picture and video capturing, their processing, analyzing and highest compressing for the following transfer in networks with low bandwidth. Mainly this work is oriented on the real-time solutions of video compression.

В наше время видео потоки высокого разрешения применяются часто и повсеместно, например, при работе с удаленным компьютером, широковещательной передаче, просмотре онлайн видеоконтента. На сегодняшнее время именно видеоданные занимают немалую часть сетевого трафика. Также видеоданные являются наиболее объемным видом хранения информации, и их передача по сети в режиме реального времени является нетривиальной задачей. В связи с этим необходимо решение проблемы эффективной передачи видео потоков больших разрешений по сетям с относительно невысокой пропускной способностью.

Данная проблема включает в себя следующие задачи: захват видео (с монитора пользователя) либо использование предоставленного видео ролика, его анализ, обработка и сжатие. Приоритетной целью является существенное сжатие видеоданных, так как объем передаваемых данных может измеряться в десятках мегабайт за секунду, что позволительно исключительно для высокоскоростных каналов.

Прежде всего, четко поставим задачу: необходимо обеспечить сжатие видео в режиме реального времени, возможна небольшая задержка при начале передачи. В цифровом виде видеоданные могут быть представлены в форме последовательности кадров. Такое представление накладывает некоторые ограничения, так как некоторые алгоритмы сжатия видео потока требуют несколько, а то и десятки кадров для успешной работы. Важным параметром является симметричность примененного метода, то есть относительно равные скорости сжатия и декодирования видео потока. По этим условиям были отброшены ряд алгоритмов сжатия. Таким образом, нужно обеспечивать эффективное сжатие, используя не более 10 – 30 кадров (приемлемая начальная задержка при передаче), и симметричностью, близкой к единице.

Для решения задачи передачи видео потока предлагается использование следующих компонентов: драйвер виртуального видеоадаптера; DLL – сборка, написанная на языке C или C++; .NET приложение. Драйвер виртуального видеоадаптера обеспечивает копирование (mirroring) данных, посылаемых реальным (физическим) видеоадаптером для отображения на мониторе. DLL – сборка представляет собой реализацию логики по работе с драйвером, осуществляет захват видео, обеспечивает логику анализа, обработки и сжатия данных. Приложение, разработанное на платформе .NET, предоставляет функциональность пользовательского интерфейса.

Драйвер виртуального видеоадаптера, работающий в режиме ядра, представляет собой удобное средство для быстрого захвата видео. Стандартные API – функции не позволяют сделать это столь же быстро, поэтому его применение намного предпочтительнее, хотя и несколько более сложное в использовании на практике.

DLL – сборка предоставляет логику работы с драйвером, осуществляет анализ, обработку и сжатие видеоданных. Написанная на языке С++, данная сборка обеспечивает более быстрое сжатие данных, в отличие от прошлой версии, где функциональность сжатия была реализована в приложении.

Важное замечание: разрабатываемое приложение является стандартным приложением Win32, и не является службой. В связи с этим, приложение будет выполняться в пользовательской терминальной сессии, поэтому при выходе пользователя из своей учетной записи приложение завершает свою работу. В дальнейшем, приложение будет изменено с учетом возможности захвата видеоданных в различных терминальных сессиях, что может быть применено для выполнения каких – либо задач различными пользователями, или при смене пользователя. Для этого будет реализована служба, взаимодействующая с драйвером виртуального видеоадаптера, эти компоненты будут предоставлять логику захвата и поддержки передачи данных вне пользовательской терминальной сессии.

При реализации метода с помощью DLL – сборки вычислялась межкадровая разница; затем полученные данные разбивались на блоки, размером 16 на 16 пикселей, и сжимались с помощью алгоритма RLE. Данный алгоритм прост в написании, быстр и симметричен.

Применение вышеописанного подхода по сжатию потоковых данных в реальном времени позволило решить поставленную задачу при помощи быстрого захвата видео с помощью драйвера, а также логики сжатия, написанной на языке С++, что обеспечило ее быстродействие. Применение алгоритма RLE также более чем оправдано: после этапа вычисления межкадровой разницы образуется большое количество повторяющихся цепочек байт, на которых алгоритм наиболее эффективен.

Использовались следующие технологии: платформа .NET + P/Invoke, также использовался виртуальный драйвер видеоадаптера. В качестве среды разработки программного обеспечения использовалась MS Visual Studio 2008, средства разработки драйверов MS Windows Driver Development Kit 6001.18001.

**REFERENCES:**

1. Гиренко А. В., Ляшенко В. В., Машталир В. П., Путятин Е. П. Методы корреляционного обнаружения объектов.– Харьков: АО “БизнесИнформ”, 1996. – 112 с.