

ПРОГРАММНАЯ СИСТЕМА ДЛЯ РАЗРАБОТКИ АЛГОРИТМОВ ВИДЕОПРОЦЕССОРА

Е.В. Калабухов, М.М. Татур

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Беларусь*

E-mail: tatur@i-proc.com; kalabukhov@bsuir.by

1. Задачи создания системы

Система видеообработки может быть построена с использованием как универсальных, так и специализированных вычислителей – видеопроцессоров. Аппаратная часть специализированной системы обычно включает в себя видеокамеру, видеопроцессор и монитор (или видеосервер).

Кроме создания собственно аппаратной части видеопроцессора, обычно требуется создавать прошивку выполняющую преобразование видеопотока, получаемого с видеокамеры, для достижения некоторого эффекта (например, повышения контрастности, резкости, выполнения фильтрации и т.п.).

Выполнение создания и отладки таких прошивок напрямую на видеопроцессоре весьма долгий процесс (создание – прошивка – тестирование – анализ – модернизация и т.д. по циклу), отнимающий много времени и требующий в ряде случаев стабильных входных видеоданных.

Кроме того создание прошивок для ряда видеопроцессоров зависит от используемой аппаратной платформы, (например, ресурсов, возможностей реализации операций обработки), хотя сам алгоритм обработки остается практически общим.

Наш подход состоит в использовании программной системы для начального создания и отладки идеи алгоритма. При этом этот этап не использует специализированный видеопроцессор, хотя формирование алгоритмов должно вестись с учетом особых требований по формату данных и допустимых операций с ними, и выполняется на персональном компьютере. В такой программной системе исходные и получаемые данные хранятся в видеофайлах (контейнеры типа .avi). Все это позволяет упростить разработку и тестирование алгоритмов, позволяет накапливать базу общих алгоритмов, которые потом можно быстрее адаптировать под нужную структуру видеопроцессора, позволяет проще сравнивать результаты и сложность реализации различных версий алгоритмов.

2. Структура системы

Программная система состоит из следующих основных компонент (см. рисунок 1):

- основного модуля, представляющего собой интерфейс пользователя и организующего работу с входными и выходным видеофайлами, а также с пользовательскими настройками процесса обработки;

- модулей обработки кадра; каждый модуль (эффект) представляет собой dll с фиксированным интерфейсом, но достаточно свободной реализацией в плане обработки кадра;
- компонента связи (специального фильтра DirectShow формата .ax, см. компонент 3 на рисунке 1) для обмена данными кадра между основным модулем и модулем обработки кадра для поддержки обработки видеопоследовательностей по технологии DirectShow.

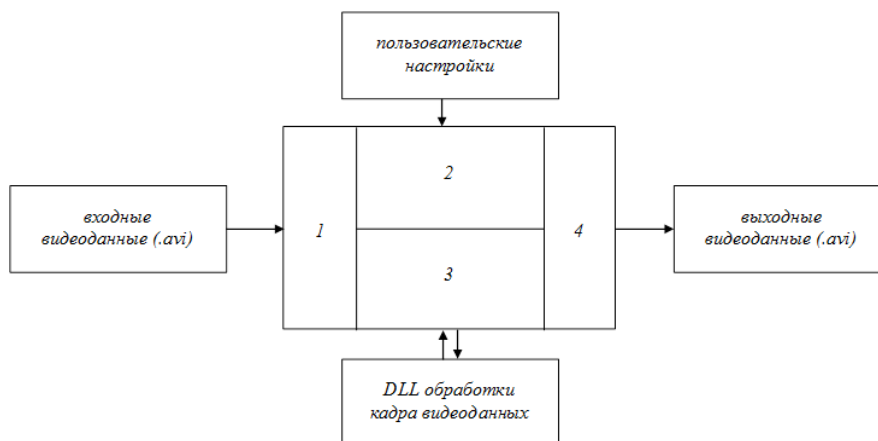


Рисунок 1. Состав программной системы (1 – интерфейс ввода видеоданных; 2 – пользовательский интерфейс; 3 – интерфейс с библиотекой обработки; 4 – интерфейс вывода видеоданных)

3. Особенности реализации системы

Система реализована на языке C++ (создавалась в MS Visual Studio 2008) для операционных систем семейства Windows (тестировалась в работе на Windows XP, 7, 8).

В основе процесса обработки видеоданных заложено использование технологии DirectShow, при этом основной программный модуль системы (см. рисунок 2) достаточно прост в использовании. Так как система предназначена для полной обработки входных видеоданных, то специализированных инструментов для выбора фрагментов видеоданных в ней не предусмотрено. Настройки системы позволяют выбрать кодек для формирования выходного видеофайла.

Эффекты для обработки выбираются из набора заранее созданных dll обработки кадров. Результаты работы одного из тестовых эффектов представлены на рисунке 3. На данный момент поддерживается обработка видеоданных только одним фильтром за проход, т.к. это позволило несколько упростить реализацию системы, хотя и снизило ее общую функциональность.

Для реализации новых модулей обработки кадров достаточно реализации dll жестко определенной структуры, которая получает программный доступ к кадру изображения. Это также одна из особенностей системы – программист отделен от прямой работы с видеопотоком и регистрации модулей эффектов в системе, что значительно упрощает создание и отладку эффектов.

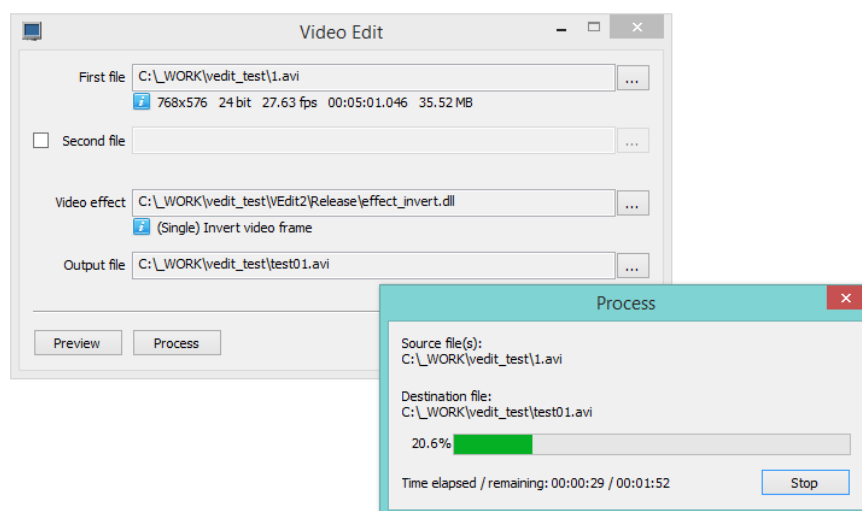


Рисунок 2. Пользовательский интерфейс программной системы

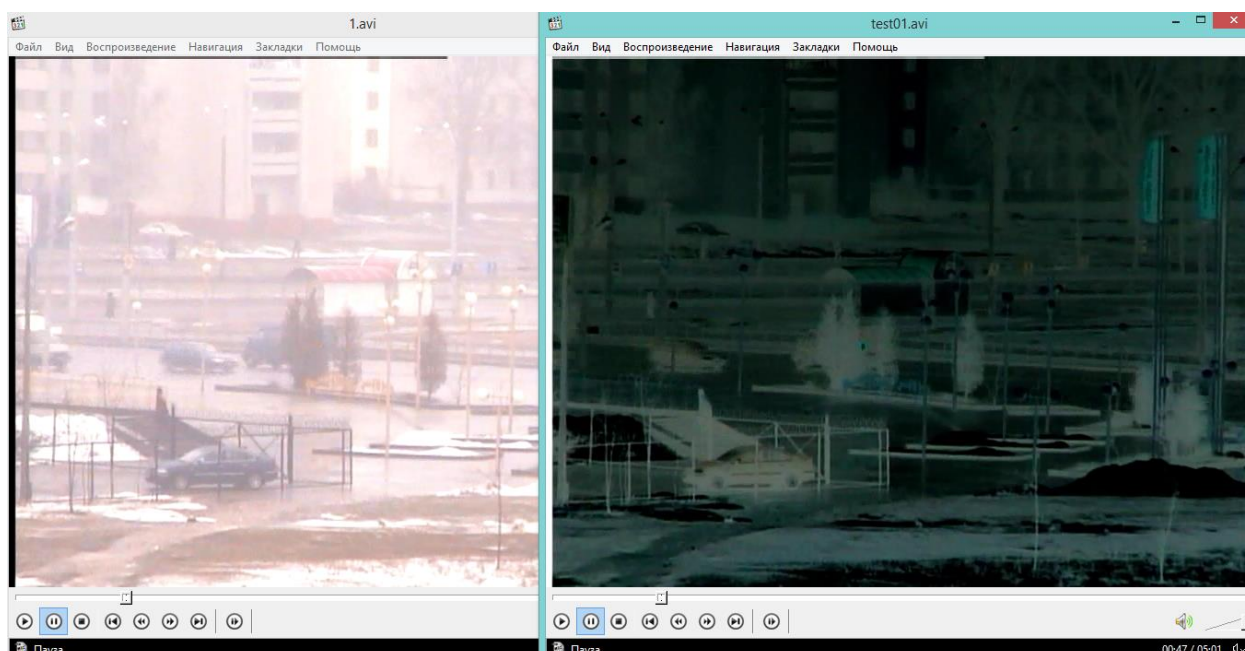


Рисунок 3. Пример работы эффекта инверсии

4. Использование системы

Полученная система позволяет:

- получить открытые в плане анализа и отладки алгоритмы обработки видеоданных, т.к. создание алгоритмов идет на достаточно низком уровне, что способствует в дальнейшем более быстрому и адекватному переводу алгоритма в прошивку видеопроцессора;
- реализовывать достаточно сложные алгоритмы обработки видеоданных, т.к. ограничения на обработку существуют только в виде договоренности о формате данных и наборе операций, которые должен соблюдать программист-создатель алгоритмов;

- хранить полученные алгоритмы в виде библиотек, при этом регистрация самих модулей обработки кадров в операционной системе не производится, что достаточно эффективно при отладке;
- анализировать результат применения алгоритмов визуальным сравнением;
- формировать видеофайлы результатов обработки, что позволяет легко демонстрировать прогресс разработки алгоритмов и проводить сравнение результатов между разными версиями одного и того же алгоритма или его настроек;
- обрабатывать как отдельные кадры, так и стерео последовательности из разных видеопотоков, что в последнем варианте может быть применимо для сравнительного анализа.

Полученная система на данном этапе в основном ориентирована на предварительную обработку изображений, однако может быть дополнена и другими технологиями анализа видеоданных, т.к. это зависит только от работы модуля обработки кадров.

Система в настоящий момент используется для создания алгоритмов нелинейной фильтрации видеоданных, которые будут в дальнейшем реализованы в прошивке оригинального видеопроцессора, построенного на базе ПЛИС Xilinx.

В дальнейшем планируется создание sdk для упрощения работы программиста с изображением кадра при создании модуля эффекта, создание библиотеки специализированных модулей для формирования статистики по обработанным видеоданным, а также доработка системы по использованию сразу нескольких модулей обработки кадров в режиме последовательной обработки видеоданных.