

Широкоугольное изображение представляет собой снимок (кадр), полученный с помощью объектива с коротким фокусным расстоянием. Преимуществом таких изображений, является большой угол зрения, однако они имеют и недостаток – дисторсию. Это означает что, линии объекта, которые являлись прямыми, искажаются, за исключением тех, которые лежат в плоскости оптической оси объектива, а величина искажения нелинейно зависит от параметров объекта. Изображения с дисторсией не пригодны для анализа ввиду того, что нарушается геометрическое подобие исходного объекта и его изображения, в связи с чем требуется предварительное исправление дисторсии.

При наличии только сферических искажений, исправить дисторсию можно зная положение точки на изображении и радиус кривизны линзы [1]. Если рассматривать изображение, в котором присутствует суперпозиция искажений необходимо предварительно рассчитать суммарную дисторсию для тестового изображения [2]. В данном случае, оно основано на определении искажений шахматной доски (либо квадратной сетки) по двум ортогональным осям (рис. 1).

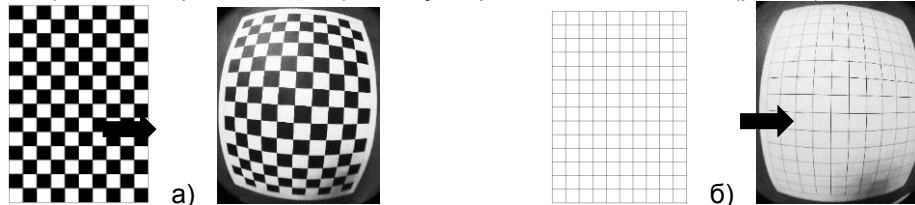


Рисунок 1 – Идеальное и реальное изображение шахматной доски (а) и сетки (б) для калибровки

Для вычисления дисторсии необходимо определить точку идеального объекта, в которой должно находиться изображение по законам оптики. В этом случае, некоторая точка реально объекта изображается в виде точки $A'(x', y')$, не совпадающей с точкой идеального изображения $A'_0(x'_0, y'_0)$. В таком случае дисторсия является отклонением точки идеального изображения A'_0 от точки реального изображения A' . Величина дисторсии ($\Delta x'$ и $\Delta y'$), в общем случае, является различной для различных точек изображения и представляет собой некоторую функцию координат точек изображения. Массивы чисел ($\Delta x', \Delta y'$) позволяют получить нелинейную зависимости дисторсии от реальных координат узловых точек изображения шахматной доски (сетки). Для этого необходимо:

- восстановить контур шахматной доски (квадратной сетки);
- определить отклонение узловых точек реального изображения от идеального;
- по величине отклонений определить вид функции изменения дисторсии;
- произвести интерполяцию значения дисторсии для точек, отличных от узловых.

Таким образом, с использованием полученной суммарной дисторсии для каждой точки тестового изображения, можно произвести коррекцию реального изображения другого объекта. Для устранения дисторсии (получения изображения, близкого к идеальному) необходимо к координатам (x', y') каждой точки A' реального изображения добавить величину дисторсии ($\Delta x', \Delta y'$).

После коррекции искажений, края изображения могут стать изогнутыми, в результате чего может потребоваться кадрирование исправленного изображения. Кроме того, после устранения дисторсии может быть перераспределено разрешение изображения: при положительной дисторсии («подушка») резкость изображения по краям может увеличиться по отношению к центру, при отрицательной дисторсии («бочка») – наоборот, резкость по краям может уменьшиться.

Список использованных источников:

1. Капустин, В.В. Коррекция координатных искажений в телевизионно-вычислительных системах / В.В. Капустин // Доклады ТУСУРа, № 4 (38), декабрь 2015 – Томск, 2015. – С. 174–177.
2. Жимбуева, Л.Д. Метод определения суммарной дисторсии цифровых изображений / Л.Д. Жимбуева // Компьютерная Оптика, том 35, №3. – Самара, 2011. – С. 347-355.

СИСТЕМА ПРОФИЛИРОВАНИЯ ДЛЯ ОПЕРАЦИОННОЙ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ

Оверченко А.С.

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

Сапронова Ю.И. – ассистент

В статье рассмотрены основы систем инструментирования и операционных систем реального времени. Начата разработка собственной системы профилирования и сбора информации о событиях системы с использованием веб-интерфейса для операционной системы реального времени TI-RTOS.

Сегодня встраиваемые системы встречаются практически повсеместно. Встраиваемая система – это система, которая выполняет определенный производителем набор функций [1]. Изменение набора этих функций доступно лишь производителю устройства, что накладывает определенный отпечаток при разработке программного обеспечения для них. Также к таким системам обычно выдвигаются высокие временные требования и требования отказоустойчивости. Это приводит как к увеличению трудозатрат на разработку таких систем, так и к использованию специализированных операционных систем, которые разработаны для работы в режиме реального времени.

Операционная система реального времени – это операционная система (ОС), которая предназначена для обслуживания многопоточных приложений, которые обрабатывают данные по мере их поступления [2]. Обычно ядро такой ОС состоит из специально спроектированного планировщика задач и примитивов синхронизации. Также к базовым компонентам таких ОС относят драйверы используемых периферийных устройств, сетевой стек и файловые системы. Но они являются опциональными и добавляются только по мере необходимости.

Инструментирование программного обеспечения связано с набором возможностей, предоставляемых для измерения производительности программного продукта, поиска компонентов, ограничивающих скорость его работы, а также записи информации трассировки и нахождения некорректно работающих модулей [3]. В инструментировании выделяют следующие направления:

- трассировка кода;
- отладка и обработка исключений;
- профилирование;
- подсчет производительности;
- логирование данных.

Стоит отметить, что обычно вышеперечисленные направления взаимосвязаны и реализуются в виде набора инструментов для разработки.

Профилирование является способом анализа программного обеспечения во время его выполнения и позволяет измерять различные параметры работы программы. Под подсчетом производительности понимают процесс предоставления информации о качестве работы разрабатываемого программного продукта с помощью счетчиков производительности. Логирование – это процесс сбора и хранения произошедших в системе событий.

К программному обеспечению, предназначенному для инструментирования, выдвигается одно важное требование: минимальное влияние на анализируемое программное обеспечение. Это делается для того, чтобы уменьшить расхождение работы программы при использовании средств инструментирования и без них.

Встречаются случаи, когда конечному пользователю также необходимо использовать средства профилирования для встраиваемой системы. Использовать классические средства профилирования, которые используются при разработке таких систем, не является возможным. Но разработчики могут предусмотреть этот вариант и добавить в систему возможность отладки с использованием специально разработанного программного обеспечения.

Эта задача и побудила автора для разработки собственной системы профилирования и логирования. Разрабатываемая система подразумевает использование ОС реального времени RTOS и предназначена для использования во встраиваемых системах с возможностью подключения с помощью Ethernet или Wi-Fi. Информация для конечного пользователя выводится на страницу веб-браузера и отображает как общую загрузенность для каждого ядра используемого микропроцессора, так и процент процессорного времени, используемого каждой задачей.

Список использованных источников:

1. Heath, Steve. Embedded Systems Design / Steve Heath. — ELSEVIER, 2003. — P. 2-8.
2. RTOS (Real-Time Operating System) — Национальная библиотека им. Н. Э. Баумана [Электронный ресурс]. – Режим доступа: [https://ru.bmstu.wiki/RTOS_\(Real-Time_Operating_System\)](https://ru.bmstu.wiki/RTOS_(Real-Time_Operating_System)). – Дата доступа: 23.03.2019г.
3. Source code instrumentation overview [Электронный ресурс]. — Режим доступа : https://www.ibm.com/support/knowledgecenter/SSSHUF_8.0.0/com.ibm.rational.testrt.doc. – Дата доступа: 12.03.2019г.