

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
Санкт-Петербургский горный университет

Кафедра автоматизации технологических процессов
и производств

МЕТОДЫ И АЛГОРИТМЫ ОБРАБОТКИ
СИГНАЛОВ И ИЗОБРАЖЕНИЙ

Методические указания к практическим работам
для студентов магистратуры направления 15.04.04

САНКТ-ПЕТЕРБУРГ
2019

УДК 004.7 (073)

МЕТОДЫ И АЛГОРИТМЫ ОБРАБОТКИ СИГНАЛОВ И ИЗОБРАЖЕНИЙ: Методические указания к практическим работам / Санкт-Петербургский горный университет. Сост.: *А.В. Бойков, А.А. Виноградова*. СПб, 2019. 30 с.

Содержат краткие теоретические сведения, информацию о порядке выполнения работ, основные требования к содержанию отчетов.

Предназначены для студентов магистратуры направления 15.04.04 «Автоматизация технологических процессов и производств».

Научный редактор проф. *В.Ю. Бажин*

Рецензент канд. техн. наук *В.В. Васильев* (ООО «ТОМС инжиниринг»)

ВВЕДЕНИЕ

Машинное зрение следует рассматривать как комплексную и технологическую область научных и инженерных знаний, охватывающую все проблемы разработки практических систем: выбор схем освещения исследуемой сцены, выбор характеристик датчиков, их количества и геометрии расположения, вопросы калибровки и ориентирования, выбор или разработка оборудования для оцифровки и процессорной обработки, разработка собственно алгоритмов и их компьютерная реализация - то есть весь круг сопутствующих задач.

Машинное зрение имеет огромное число потенциальных областей применения, таких как промышленная инспекция и контроль качества, робототехника, навигация и транспортировка, дистанционное зондирование, медицина и биомеханика, инженерный труд, автоматизация проектирования, новые технологии обработки документов, биометрия и множество других. Мы будем использовать его для организации зрительной обратной связи при работе управляемых устройств, манипуляторов или мобильных роботов в изменчивой среде.

Основные задачи машинного зрения могут быть сформулированы следующим образом:

- 1) обнаружение объектов и изменений в сцене наблюдения;
- 2) описание сцены и идентификация объектов;
- 3) слежение за объектами;
- 4) калибровка сенсоров, самоориентация и самопозиционирование;
- 5) реконструкция поверхностей и обнаружение трехмерных структур;
- 6) высокоточные измерения элементов сцены;
- 7) организация зрительной обратной связи при работе управляемых устройств, манипуляторов или мобильных роботов в изменчивой среде.

ПРАКТИЧЕСКАЯ РАБОТА № 1. ГЕОМЕТРИЧЕСКИЕ И АЛГЕБРАИЧЕСКИЕ ПРЕОБРАЗОВАНИЯ ИЗОБРАЖЕНИЙ

Цель работы: изучение геометрические и алгебраические преобразования изображений в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Большинство изображений с физической точки зрения представляет собой зарегистрированное специальным датчиком двумерное распределение интенсивности электромагнитного излучения, отраженного объектом регистрации. Между тем все, что нам на самом деле дано в двумерном изображении, - это видимое поле, представляющее собой лишь некоторую функцию распределения яркости или цвета на двумерной плоскости: $f(x, y)$, где x и y - декартовы координаты, описывающие плоскость изображения. В качестве детектора используют ПЗС-матрицу. Она формирует 2D-массив значений в каждой точке изображения, пикселе. Изображения, состоящие из значений пикселей, называют растровыми.

Начиная изучать машинное зрение, необходимо четко представлять себе, что в компьютере в качестве цифрового изображения хранится только и исключительно двумерный массив. Любые другие данные, которые мы хотели бы из изображения извлечь (фигуры, линии, объекты, размеры, содержание изображенного текста и т. д. и т. п.) - могут быть получены лишь в результате применения ряда процедур обработки и анализа изображения, которые мы должны либо сами запрограммировать.

Рассмотрим для начала алгебраические операции, которые осуществляются над цифровыми изображениями как над двумерными матрицами. Здесь предполагается, что мы имеем два цифровых изображения $Im1$ и $Im2$ одного типа и одного размера $DimX \times DimY$. В этом случае мы можем осуществить над находящимися в одинаковых позициях этих изображений пикселями соответственно любые арифметические или логические операции, которые в принципе могут быть осуществлены над двумя числами. Изображение того же типа и размера будет сформировано в результате применения такой попиксельной операции.

Функции IMAQ: Палитра Operators. Палитра Operators содержит функции, реализующие попиксельные алгебраические операции над растровыми изображениями как двумерными массивами данных. Список алгебраических операций, реализованных в палитре Operators, перечислен ниже в табл. 1.1.

Таблица 1.1

Алгебраические операции над растровыми изображениями из палитры Operators

N	Название функции	Операция над пикселями	Смысл операции
1	IMAQ Absolute Difference	$Im3[x,y] = \text{abs}(Im1[x,y] - Im2[x,y])$	Модуль разности
2	IMAQ Add	$Im3[x,y] = Im1[x,y] + Im2[x,y]$	Сумма
3	IMAQ And	$Im3[x,y] = Im1[x,y] \text{ and } Im2[x,y]$	Побитовое логическое И
4	IMAQ Compare	$Im3[x,y] = \text{op}(Im1[x,y] > Im2[x,y])$	Сравнение изображений. Если условие справедливо, применяется операция op, выбранная из списка
5	IMAQ Divide	$Im3[x,y] = Im1[x,y] / Im2[x,y]$	Деление
6	IMAQ LogDiff	*** (см. описание ниже)	Логическая побитовая разность
7	IMAQ Mask	$Im3[x,y] = Im1[x,y] \text{ and } (Im2[x,y] > 0)$	«Маскирование»: в выходное изображение передаются только те пиксели исходного, которые соответствуют ненулевым пикселям изображения-маски
8	IMAQ Modulo	$Im3[x,y] = Im1[x,y] \text{ mod } Im2[x,y]$	Остаток целочисленного деления
9	IMAQ MulDiv	$Im3[x,y] = (Im1[x,y] \times \text{Const}) / Im2[x,y]$	Умножение на константу и деление на изображение
10	IMAQ Multiply	$Im3[x,y] = Im1[x,y] \times Im2[x,y]$	Произведение
11	IMAQ Or	$Im3[x,y] = Im1[x,y] \text{ or } Im2[x,y]$	Побитовое логическое ИЛИ
12	IMAQ Subtract	$Im3[x,y] = Im1[x,y] - Im2[x,y]$	Разность
13	IMAQ Xor	$Im3[x,y] = Im1[x,y] \text{ xor } Im2[x,y]$	Побитовое логическое исключающее ИЛИ

Функция IMAQ: Absolute Difference осуществляет попиксельное вычитание изображений. Результат операции берется по модулю.

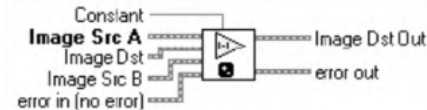


Рис. 1.1 Функция IMAQ: Absolute Difference

Пример работы данной функции представлен на рис. 1.2. Блок-диаграмма соответствующей схемы обработки представлена на рис. 1.3.



Рис. 1.2 Результат работы функции Absolute Difference

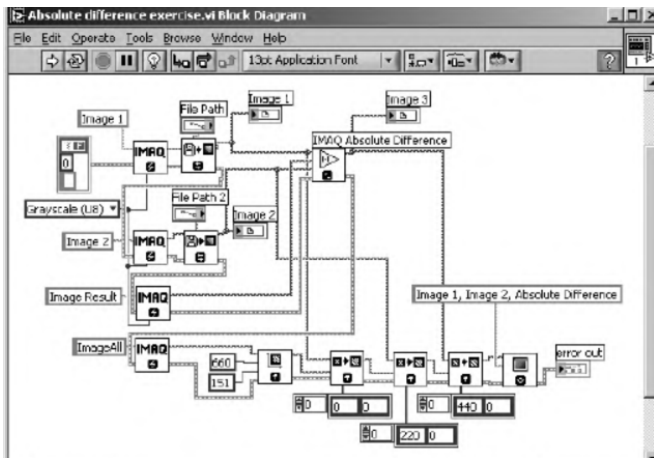


Рис. 1.3 Блок-диаграмма, демонстрирующая работу Absolute Difference

Задание на практическую работу:

Изучить работу следующих функций IMAQ: Add, And, Compare, LogDiff, Modulo, MulDiv, Or, Subtract, Xor.

Собрать схему, демонстрирующую возможности каждой функции. Оформить отчет, содержащий программный код и результат его работы с комментариями и пояснениями.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКАЯ РАБОТА № 2. ФОРМАТЫ ХРАНЕНИЯ, СЖАТИЯ И ПЕРЕДАЧИ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ И ВИДЕО

Цель работы: изучение форматов изображений и реализация работы алгоритма сжатия изображений, основанного на стандарте JPEG в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Рассмотрим растровое цифровое изображение как чистую структуру данных. Поскольку с геометрической (и программной) точки зрения структура любого изображения совершенно стандарт-

на и представляет собой регулярный двумерный массив (матрицу) пикселей, различные типы изображений могут порождаться только оригинальным типом данных, представляющим пиксель. Наиболее распространенными на сегодняшний день являются следующие форматы цифровых изображений: BMP, PCX, TIFF, GIF, JPEG.

Таблица 2.1

Файловые форматы цифровых изображений

Формат	Макс. число бит/пиксель	Макс. число цветов	Макс. размер изображения, пиксель	Метод сжатия	Хранение нескольких изображений
BMP	24	16*777*216	65535*65535	-, RLE *	-
PCX	24	16*777*216	65535*65535	RLE	-
GIF	8	256	65*535*65535	LZW	+
TIFF	24	16*777*216	всего 4*294*967*295	-, LZW, RLE, др.*	+
JPEG	24	16*777*216	65535*65535	JPEG	-

Как видно, одной из важных характеристик формата цифрового изображения является применяемый в данном формате метод сжатия. Методы сжатия данных делятся на две основные группы: сжатие без потерь и сжатие с потерями информации.

JPEG – (Joint Photographic Experts Group – название организации-разработчика) метод сжатия изображений с потерями, разработанный группой экспертов в области фотографии для сжатия полноцветных изображений. Разработаем программу, выполненная в среде LabVIEW, которая включает в себя графический интерфейс и блок диаграмму, которая описывает следующий алгоритм:

1) *Преобразование цветного изображения в матрицу, содержащую оттенки серого.* В блоке преобразования изображения в матрицу происходит загрузка изображения и его преобразование в матрицу значений оттенков серого, где значению каждой ячейки соответствует значение яркости пиксела от 0 до 255. Загрузка изображения осуществляется блоком IMAQ «ReadFile». Блок IMAQ «ImageToAlpha» преобразует изображение в двумерный массив, с возможностью представления числа в формате 8 бит, 16 бит или с плавающей точкой, в зависимости от типа входного изображения.

2) *Прямое дискретное косинусное преобразование (ДКП).* С помощью ДКП осуществляется переход от пространственного представления изображения к спектральному представлению и обратно.

Таблица 2.2

Матрица дискретного косинусного преобразования

j	i							
	1	2	3	4	5	6	7	8
1	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$	$\frac{1}{\sqrt{8}}$
2	$\frac{1}{2} \cos \frac{\pi}{16}$	$\frac{1}{2} \cos \frac{3\pi}{16}$	$\frac{1}{2} \cos \frac{5\pi}{16}$	$\frac{1}{2} \cos \frac{7\pi}{16}$	$\frac{1}{2} \cos \frac{9\pi}{16}$	$\frac{1}{2} \cos \frac{11\pi}{16}$	$\frac{1}{2} \cos \frac{13\pi}{16}$	$\frac{1}{2} \cos \frac{15\pi}{16}$
3	$\frac{1}{2} \cos \frac{2\pi}{16}$	$\frac{1}{2} \cos \frac{6\pi}{16}$	$\frac{1}{2} \cos \frac{10\pi}{16}$	$\frac{1}{2} \cos \frac{14\pi}{16}$	$\frac{1}{2} \cos \frac{18\pi}{16}$	$\frac{1}{2} \cos \frac{22\pi}{16}$	$\frac{1}{2} \cos \frac{26\pi}{16}$	$\frac{1}{2} \cos \frac{30\pi}{16}$
4	$\frac{1}{2} \cos \frac{3\pi}{16}$	$\frac{1}{2} \cos \frac{9\pi}{16}$	$\frac{1}{2} \cos \frac{15\pi}{16}$	$\frac{1}{2} \cos \frac{21\pi}{16}$	$\frac{1}{2} \cos \frac{27\pi}{16}$	$\frac{1}{2} \cos \frac{33\pi}{16}$	$\frac{1}{2} \cos \frac{39\pi}{16}$	$\frac{1}{2} \cos \frac{45\pi}{16}$
5	$\frac{1}{2} \cos \frac{4\pi}{16}$	$\frac{1}{2} \cos \frac{12\pi}{16}$	$\frac{1}{2} \cos \frac{20\pi}{16}$	$\frac{1}{2} \cos \frac{28\pi}{16}$	$\frac{1}{2} \cos \frac{36\pi}{16}$	$\frac{1}{2} \cos \frac{44\pi}{16}$	$\frac{1}{2} \cos \frac{52\pi}{16}$	$\frac{1}{2} \cos \frac{60\pi}{16}$
6	$\frac{1}{2} \cos \frac{5\pi}{16}$	$\frac{1}{2} \cos \frac{15\pi}{16}$	$\frac{1}{2} \cos \frac{25\pi}{16}$	$\frac{1}{2} \cos \frac{35\pi}{16}$	$\frac{1}{2} \cos \frac{45\pi}{16}$	$\frac{1}{2} \cos \frac{55\pi}{16}$	$\frac{1}{2} \cos \frac{65\pi}{16}$	$\frac{1}{2} \cos \frac{75\pi}{16}$
7	$\frac{1}{2} \cos \frac{6\pi}{16}$	$\frac{1}{2} \cos \frac{18\pi}{16}$	$\frac{1}{2} \cos \frac{30\pi}{16}$	$\frac{1}{2} \cos \frac{42\pi}{16}$	$\frac{1}{2} \cos \frac{54\pi}{16}$	$\frac{1}{2} \cos \frac{66\pi}{16}$	$\frac{1}{2} \cos \frac{78\pi}{16}$	$\frac{1}{2} \cos \frac{90\pi}{16}$
8	$\frac{1}{2} \cos \frac{7\pi}{16}$	$\frac{1}{2} \cos \frac{21\pi}{16}$	$\frac{1}{2} \cos \frac{35\pi}{16}$	$\frac{1}{2} \cos \frac{49\pi}{16}$	$\frac{1}{2} \cos \frac{63\pi}{16}$	$\frac{1}{2} \cos \frac{77\pi}{16}$	$\frac{1}{2} \cos \frac{91\pi}{16}$	$\frac{1}{2} \cos \frac{105\pi}{16}$

Поскольку вычисление ДКП весьма трудоемко, вместо анализа всего изображения сразу, его разбивают на матрицы размером 8x8 элементов и затем вычисляют ДКП для каждой матрицы. В блоке разбиения на матрицы размером 8x8 элементов высчитывается количество блоков 8x8, на которые может быть разбито загруженное изображение и происходит разбиение изображения. В блоке дискретного косинусного преобразования над каждым блоком производится ДКП по следующей формуле: $T=V M V^T$, где V (элементы матрицы исходного изображения), T (элементы матрицы преобразованного изображения), M (Матрица дискретного косинусного преобразования) - матрицы размером 8x8.

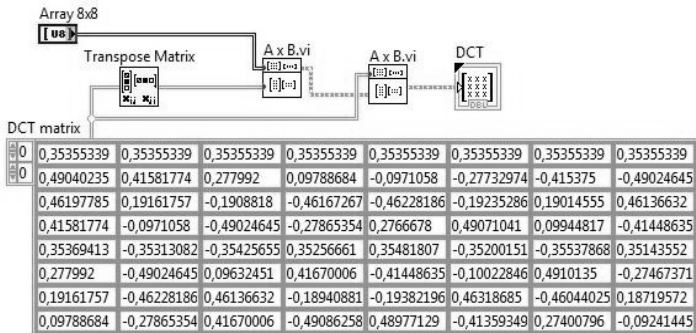


Рис. 2.1 Схема подпрограммы, выполняющей ДКП

3) *Квантование и деквантование.* В блоке квантования каждый блок делится на матрицу квантования, рассчитанную в соответствии с заданным коэффициентом качества, и производится округление получившихся значений до ближайшего целого числа. Формула для вычисления матрицы квантования, приведенная в стандарте JPEG, выглядит следующим образом: $Q_{ij}=1+(i+j) \cdot R$, где R - коэффициент качества, который задаётся пользователем. В блоке деквантования полученный ранее массив чисел умножается на матрицу качества.

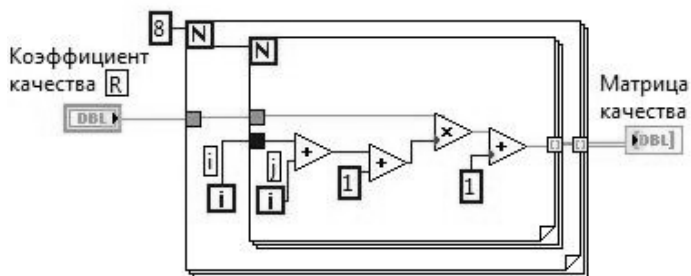


Рис. 2.2 Схема подпрограммы, вычисляющей матрицу качества

4) *Обратное дискретное косинусное преобразование (ДКП).* В блоке обратного дискретного косинусного преобразования над каждым блоком производится обратное ДКП по формуле: $V = M^T T M$.

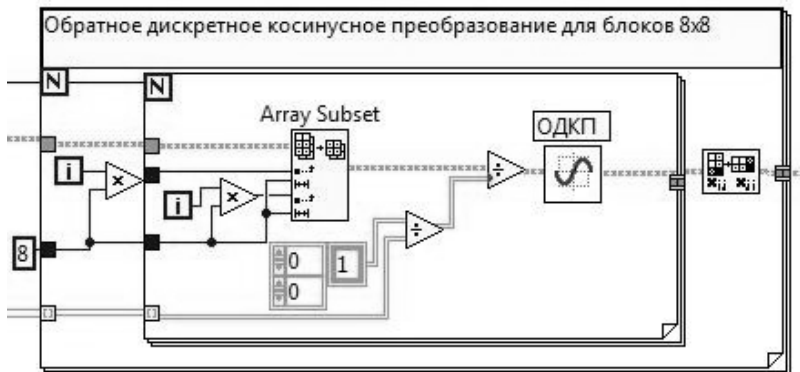


Рис. 2.3 Схема цикла «For Loop», в котором выполняется обратное ДКП

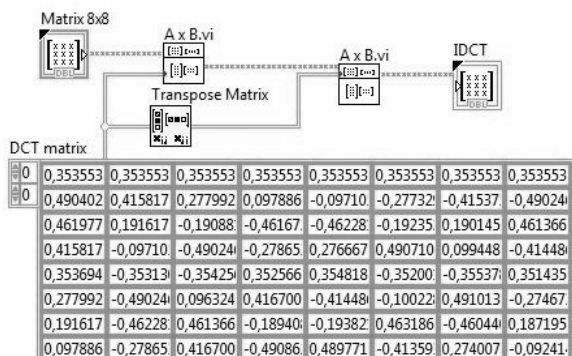


Рис. 2.4 Схема программы, выполняющей обратное ДКП

5) *Восстановление матрицы оттенков серого в цветное изображение.* В блоке соединения матриц 8x8 полученные после восстановления матрицы размером 8x8 соединяются в матрицу исходного размера. В блоке преобразования полученной матрицы в изображение из матрицы чисел восстанавливается изображение.

Вычисление ошибки. Ошибка вычисляется как среднеквадратическое отклонение разности исходного изображения и закодированного:

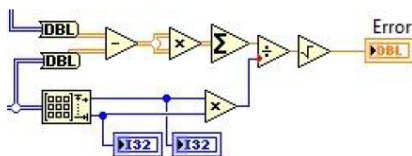


Рис. 2.5 Вычисление ошибки между исходным изображением и сжатым

б) *Сохранение изображения.*

Функция «IMAQ ArrayToImage» преобразовывает матрицу в изображение. В программе предусмотрена возможность сохранения сжатого изображения на компьютер.

Задание на практическую работу:

Реализовать алгоритм сжатия изображений JPEG в программном пакете LabVIEW с использованием средств IMAQ Vision. Изучить работу алгоритмов сжатия. Алгоритм программы представлен на рис. 2.6, а примерный вид лицевой панели на рис. 2.7.

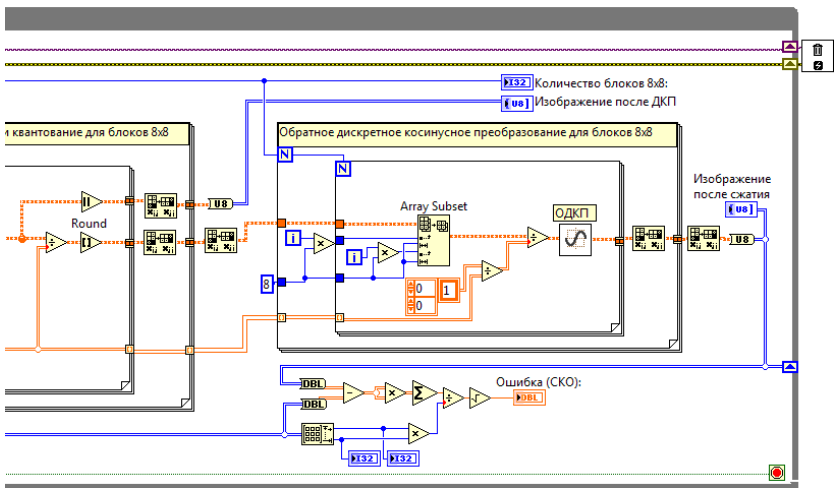
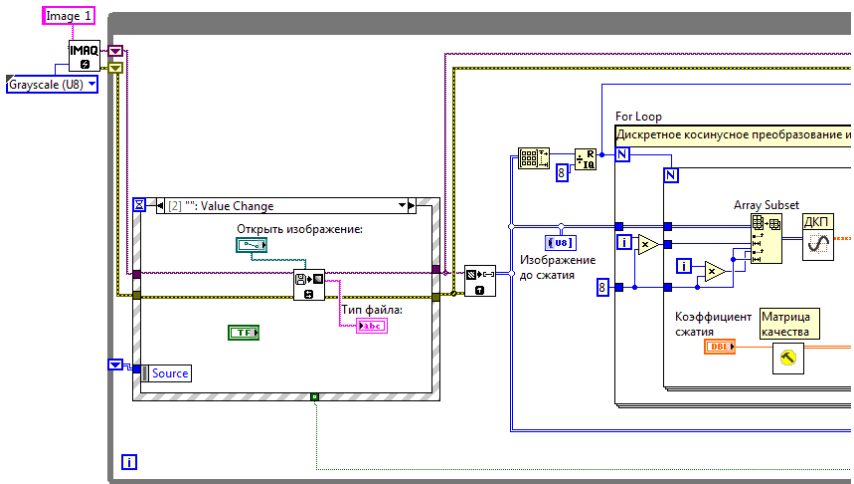


Рис. 2.7 Общий вид программы лабораторной работы «Исследование алгоритма сжатия изображений»

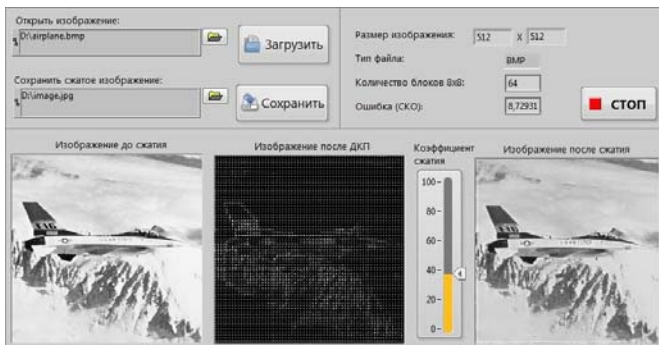


Рис. 2.6 Лицевая панель программы

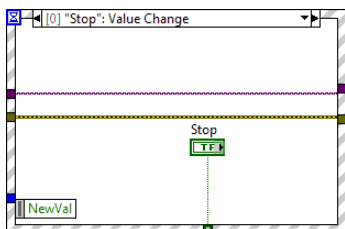


Рис. 2.8 Обработка кнопки

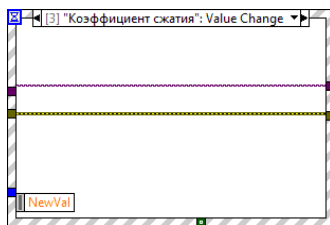


Рис. 2.9 Обработка индикатора «Степень сжатия»

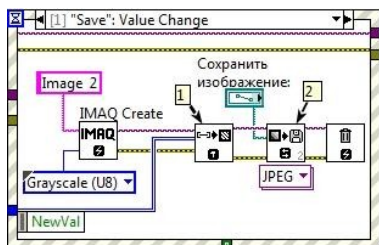


Рис. 2.10 Часть программы для сохранения изображения

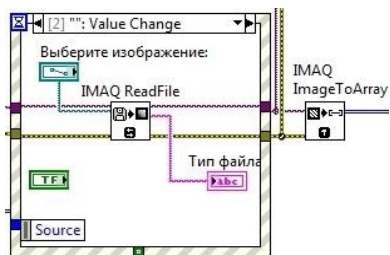


Рис. 2.11 Часть программы, осуществляющая загрузку изображения

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКАЯ РАБОТА № 3. ПЛАТФОРМА NI VISION: ЗАХВАТ ИЗОБРАЖЕНИЙ И ВИДЕО В LABVIEW

Цель работы: программирование USB-камер для захвата изображений и видеопоследовательностей в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Для программирования USB-камеры надо указать имя камеры. Перечень допустимых камер можно получить с помощью функции IMAQ USB Enumerate Cameras. Выходной параметр этой функции - строковый массив с именами и характеристиками доступных камер.

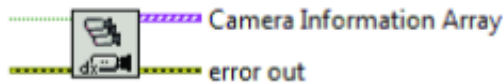


Рис. 3.1 Блок IMAQ USB Enumerate Cameras

Процесс захвата состоит из нескольких этапов:

- 1) инициализация камеры;
- 2) подготовка буфера для изображения;
- 3) запуск сессии захвата;
- 4) непосредственно захват изображения;
- 5) закрытие сессии захвата изображения.

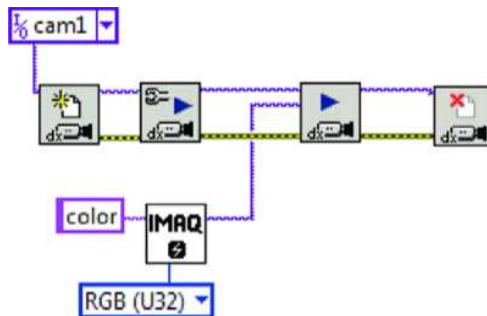


Рис. 3.2 Программа захвата изображения

Рассмотрим эти этапы подробнее:

1) *Инициализация камеры.* При подключении к камере происходит открытие соединения для работы с ней, то есть создается уникальная ссылка для этой камеры для последующего обращения при программировании к выбранной камере через эту ссылки. Эту функ-

цию в программе надо выполнять один раз для данной камеры. Из списка доступных видеокamer выберем одну, например, с именем cam0.

Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – NI-IMAQ dx – Open.

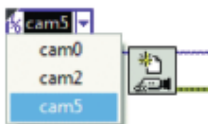


Рис. 3.3 Выбор камеры

2) *Подготовка буфера для изображения.* Буфер изображения - ссылка на область памяти, где планируется хранить изображение. Для резервирования определенным образом устроенной области памяти надо указать прежде всего имя этой области (параметр Image Name) и принцип организации хранения изображения, - то есть формат изображения (параметр Image Type).

Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – Vision Utilities – Image Management – IMAQ Create.

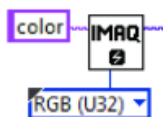


Рис. 3.4 Буфер изображения

3) *Запуск сессии захвата.* Этот процесс организует, подготавливает саму сессию захвата изображения. Эту функцию в программе надо выполнять один раз для данной камеры.

Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – NI-IMAQ dx – Configure Grab.



Рис. 3.5 Сессия захвата

4) *Непосредственно захват изображения.* Эта операция может быть выполнена в программе многократно, нужное число раз.

Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – NI-IMAQ dx – Grab.



Рис. 3.6 Захват изображения

5) Закрытие сессии захвата изображения. Эту функцию в программе надо выполнять один раз для данной камеры. Она делает недоступной работу с камерой с конкретным именем.

Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – NI-IMAQ dx – Close.



Рис. 3.7 Закрытие сессии захвата изображения

Задание на практическую работу:

Внесите изменения в программу, чтобы захват изображения с веб-камеры был многократным. Для этих целей используйте структуру цикла с выходом по условию. В качестве окончания выполнения цикла отслеживайте нажатие кнопки Stop. В цикле необходимо выполнять только непосредственно захват изображения.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКАЯ РАБОТА 4. МЕТОДЫ ОБРАБОТКИ ИЗОБРАЖЕНИЙ: БИНАРИЗАЦИЯ И ФИЛЬТРАЦИЯ ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ

Цель работы: изучение методов бинаризации и фильтрации в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Изображения называют изображениями в градациях серого,

если в них присутствуют различные оттенки серого цвета от очень темного серого - черного до совершенно светлого серого - белого. И наконец, встречаются черно-белые изображения, состоящие только из белого и черного цветов, то есть из двух цветов. Такие изображения называют бинарными. Цвета могут быть не только черный и белый, но и другие любые, например, красный и черный, но их в изображении обязательно только два.

Почти весь экспресс-анализ и обработку изображений проводят для изображений в градациях серого. Оно имеет меньший объем занимаемой памяти в компьютере, чем цветное, имеет более простую модель хранения в виде 2D-матрицы однобайтных чисел интенсивностей пикселей. Поэтому нам необходимо научиться преобразовывать зарегистрированное цветное изображение в изображение градаций серого.

Функция IMAQ Cast Image осуществляет преобразование цветного RGB-изображения в изображение градаций серого.

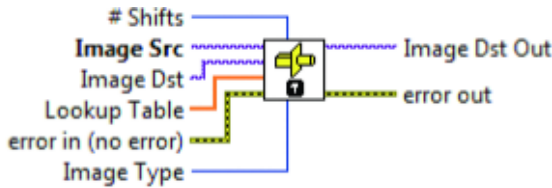


Рис. 4.1 Функция IMAQ Cast Image

Функция отображения значений пикселей изображения в 2D-массив важна для оценки количественных значений.

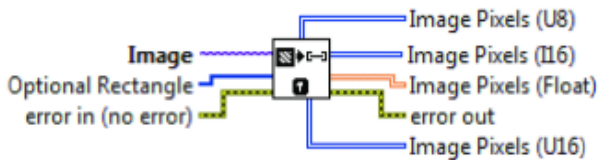


Рис. 4.2 Функция IMAQ Image To Array

Следующая программа будет осуществлять преобразование цветного изображения в оттенки серого.

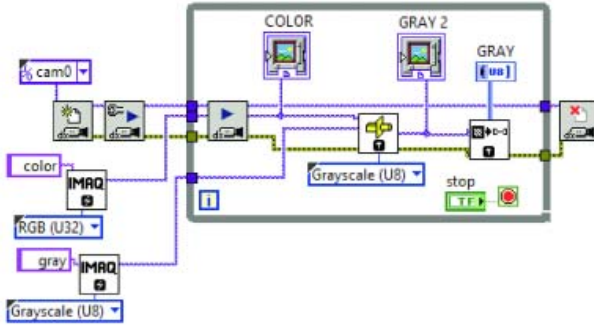


Рис. 4.3 Программа преобразования цветного изображения в оттенки серого

После запуска получим исходное зарегистрированное цветное изображение и это же преобразованное в градации серого изображение.

Продолжим процесс упрощения структуры изображения. Проведем бинаризацию изображения. Разобьем все пиксели изображения градаций серого на две области. Одна из областей содержит все пиксели со значением ниже некоторого уровня интенсивности, а другая содержит все пиксели со значением выше этого уровня интенсивности. Будем задавать уровень бинаризации эмпирическим путем, то есть методом подбора. Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – Image Processing – Processing – IMAQ Threshold.

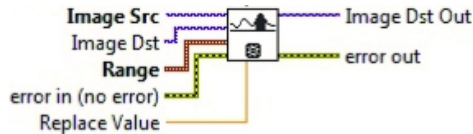


Рис. 4.1 Функция IMAQ Threshold

Задание на практическую работу:

Объединить процесс преобразования изображения из цветного в оттенки серого и бинаризации в одной программе. Результат вывести на лицевую панель в виде 2D-массива.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКАЯ РАБОТА 5. ОБРАБОТКА ЦВЕТНЫХ ИЗОБРАЖЕНИЙ

Цель работы: изучение методов обработки цветных изображений в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Существует несколько цветовых моделей изображений. Рассмотрим одну из них - цветовую модель RGB. Каждая точка-пиксель RGB - изображений представима тремя цветами: Red - красный, Green - зеленый, Blue - голубой. На хранение каждого цвета пикселя выделяется один байт. Получаем, что цвет в каждом пикселе изображения формируется из 256 оттенков красного цвета, 256 оттенков зеленого цвета и 256 оттенков голубого. Всего цветов в палитре $16\ 777\ 216 = 2^N$. Легко вычислить объем памяти, необходимый для хранения цветного RGB-изображения: количество точек в изображении умножить на 3 байта. На самом деле компьютеру удобнее работать с группами, число байтов в которых является степенью 2. Поэтому для хранения цвета одного пикселя выделяют не три, а четыре байта. Таким образом, один байт не используется, но работать многим программам с такими изображениями удобнее, быстрее. Попробуем оценить количественно цвет каждого пикселя растрового изображения. Это должно быть целое число, занимающее 3 байта памяти. Для этого воспользуемся функцией, которая находится в палитре Vision and Motion – Vision Utilities – Color Utilities – IMAQ ColorImageToArray.

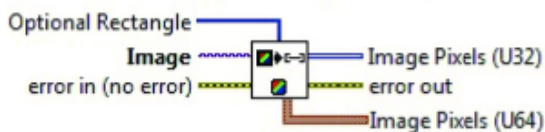


Рис. 5.1 Функция IMAQ ColorImageToArray

Создадим программу обработки цветного изображения. Используем правило размещения красного, зеленого и голубого цветов в RGB цветной модели, можно оценить значения каждого из цветов в 20-матрице числовых значений цвета пикселя изображения.

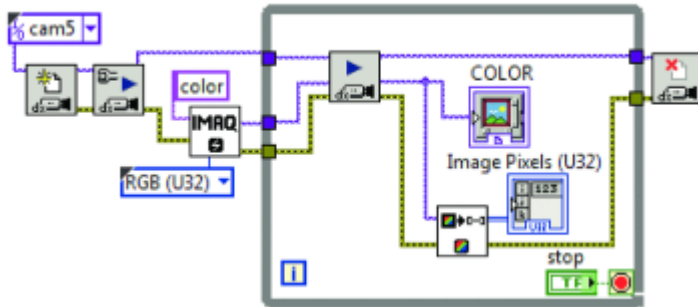


Рис. 5.2 Программа обработки цветного изображения

Создадим дополнительные буферы изображений IMAQ Create для отображения Red, Green и Blue составляющих в градациях серого для исходного цветного изображения. Разместим для этих составляющих на лицевой панели окна для отображения их в градациях серого.

В цикле поставим IMAQ Extract Planer функцию разложения цветного GRB-изображения на составляющие цветов Red, Green и Blue. Разместим также функции для отображения в 2D-массив целых чисел, составляющих трех цветов изображения. Для этого воспользуемся функцией, которая находится в палитре Vision and Motion – Vision Utilities – Color Utilities – IMAQ ExtractColorPlanes.

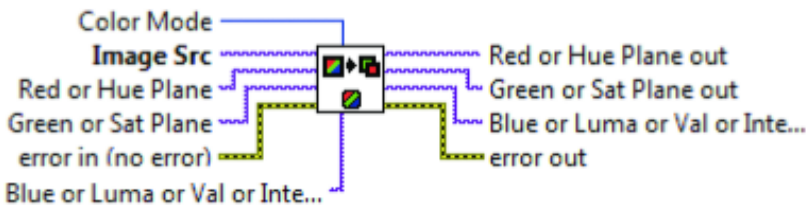


Рис. 5.3 Функция IMAQ ExtractColorPlanes

Функция IMAQ ImageToArray отображения значений пикселей изображения в 2D-массив важна для оценки количественных значений.

Программа для распознавания цвета пикселя RGB-изображения будет выглядеть следующим образом:

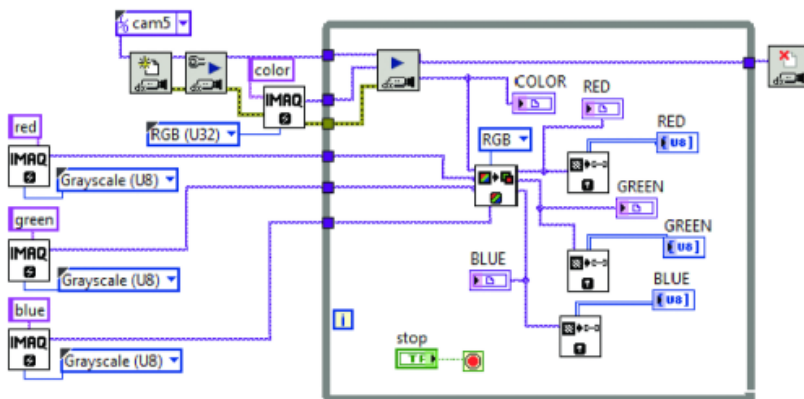


Рис. 5.4 Программа распознавания цвета пикселя

После запуска получим исходное цветное изображение и в трех окошках изображения каждой из трех составляющих цветов исходного изображения.

Задание на практическую работу:

- 1) Разложить RGB-изображение на составляющие и сделать вывод по каждому цвету в количественной форме.
- 2) Разработать программу автоматического распознавания цвета пикселя, выбранного с помощью указателя мышки на изображении.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

**ПРАКТИЧЕСКАЯ РАБОТА № 6. МЕТОДЫ АНАЛИЗА
ИЗОБРАЖЕНИЙ: ВЫДЕЛЕНИЕ И АНАЛИЗ СВЯЗНЫХ
ОБЛАСТЕЙ; ВЫДЕЛЕНИЕ КОНТУРОВ НА ПОЛУТОНОВЫХ
ИЗОБРАЖЕНИЯХ; ВЫДЕЛЕНИЕ ГЕОМЕТРИЧЕСКИХ
ПРИМИТИВОВ**

Цель работы: изучение методов выделения контуров и геометрических примитивов в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Металлические объекты на бинарном изображении, как правило, имеют сложную картину отраженного света.



Рис. 6.1 Изображение металлического объекта

На ней присутствуют блики, геометрию которых трудно предсказать априори. Желательно было бы удалить все блики с объекта программным путем. Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – Image Processing – Morphology – IMAQ Fill Hole.

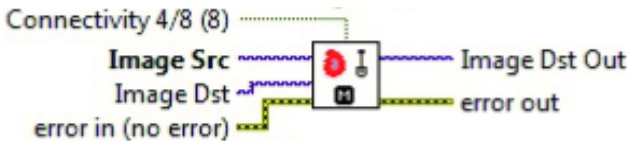


Рис. 6.2 Функция IMAQ FillHole

Эта функция в бинарном 8-битном изображении заполняет внутренние области всех замкнутых контуров значениями интенсивности пикселей 1. Таким образом, если нам априори известно, что объект имеет однородную поверхность, то можно применить данную функцию и изучать на бинарном изображении форму объектов.

Блок-диаграмма этой программы имеет вид:

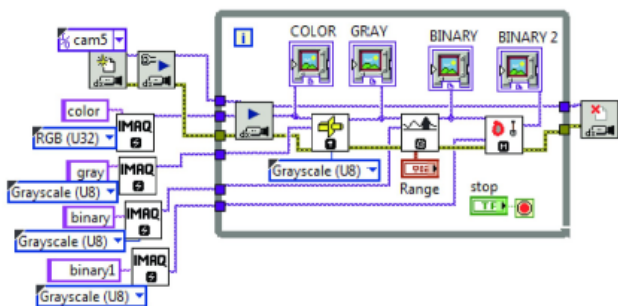


Рис. 6.3 Блок-диаграмма

На окошке индикатора BINARY 2 выведено бинарное изображение после применения функции заполнения контура. Видно, что рисунок поверхности монет полностью утерян, но четко выявлена форма объектов. Это будем использовать в следующем примере, для определения достоинства монет по их форме.

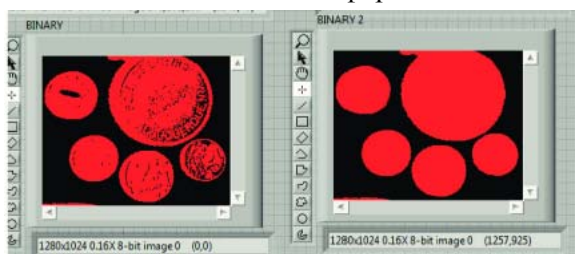


Рис. 6.4 Результат работы блок-диаграммы

Задание на практическую работу:

Распознавать монеты в сцене наблюдения. Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – Image Processing– Morphology– IMAQ Find Circles (поиск окружностей) и подсчитать их количество и номинал.

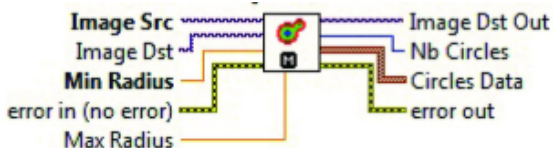


Рис. 6.5 Функция IMAQ Find Circles

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКАЯ РАБОТА № 7. ОБНАРУЖЕНИЕ ОБЪЕКТОВ, ЗАДАННЫХ ЭТАЛОНАМИ

Цель работы: изучение методов обнаружения объектов, заданных эталоном, в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

Для нахождения на сцене более сложных объектов необходимо создать для них отличительные примитивы или шаблоны. Обычно шаблоны создаются для отображений объектов на изображениях в градациях серого. Функция поиска шаблона объекта на сцене сверяет заданный фрагмент попиксельно на совпадение. Но надо уметь правильно выбирать размер шаблона в каждом конкретном случае, решая задачу оптимизации быстродействия выполнения программы и достоверности идентификации объекта: чем меньше размер шаблона, тем быстрее будет работать программа, но тем сложнее достоверно выделить наш объект на сцене.

Рассмотрим программу для создания одного шаблона с помощью мышки с записью его в файл. Для этого воспользуемся функцией, которая находится в палитре: IMAQ Select Rectangle - выделение координат области на изображении с помощью мышки и функцией, которая находится в палитре: Vision and Motion – Machin Vision – Select Region of Interest – IMAQ Select Rectangle.

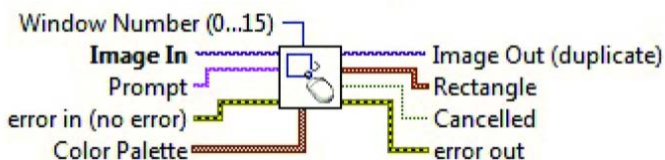


Рис. 7.1 Функция IMAQ Select Rectangle

Выделим с помощью функции IMAQ Extract из изображения область с заданными координатами. Для этого воспользуемся функцией, которая находится в палитре: Vision and Motion – Vision Utilities – Images Manipulation – IMAQ Extract.

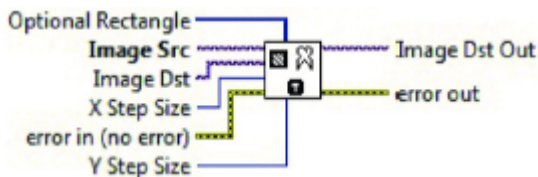


Рис. 7.2 Функция IMAQ Extract

С помощью функции IMAQ WriteFile запишем полученный фрагмент изображения в файл.

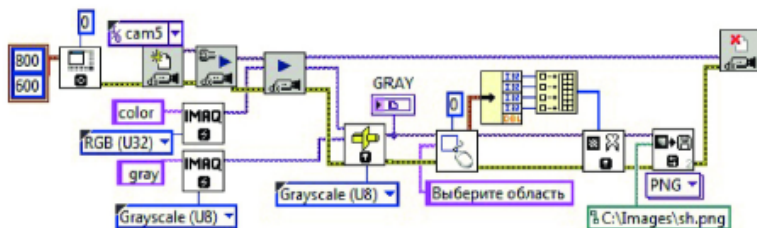


Рис. 7.3 Блок-диаграмма

Перейдем к поиску на сцене заданных объектов. Организуем на блок-диаграмме процесс захвата цветного изображения, преобразование его в изображение градаций серого и считывание из файла изображения.

Для организации поиска шаблона на сцене наблюдения воспользуемся функцией, которая находится в палитре: Vision and Motion – Machine Vision – IMAQ Find Pattern.

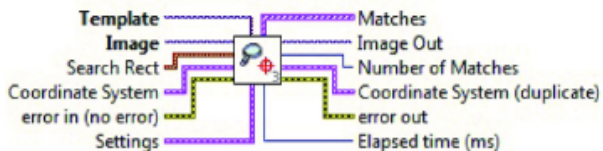


Рис. 7.4 Функция поиска шаблона

Функция IMAQ Find Pattern имеет ряд настроек:

- 1) Терминал Match Mode - выбор одного из двух возможных режимов поиска: поиск, инвариантный изменению масштаба, и поиск, инвариантный повороту шаблона.
- 2) Number of Matches Requested - число ожидаемых расположений шаблонов на сцене. Этот параметр может регулировать коли-

чество найденных фрагментов, после того как провели поиск данное число раз, поиск прекращается.

3) Minimum Match Score - наименьшее допустимое число совпадений пикселей в шаблоне.

4) Rotation Angle Ranger - массив диапазонов углов поиска.

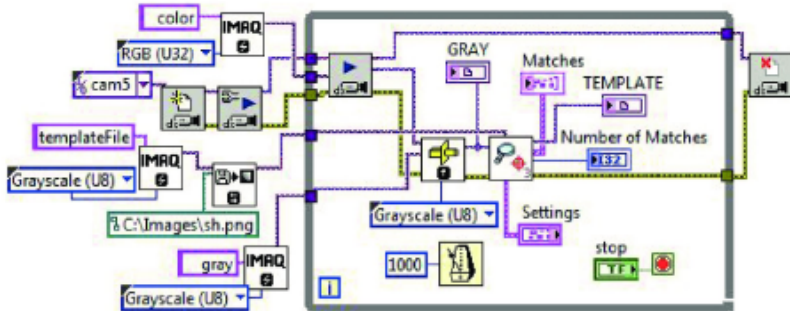


Рис. 7.5 Блок-диаграмма

Задание на практическую работу:

Показать результат работы алгоритма поиска объектов, заданных эталоном. Описать ограничения и области применения данного метода.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

ПРАКТИЧЕСКАЯ РАБОТА № 8. СЧИТЫВАНИЕ СИМВОЛЬНОЙ ИНФОРМАЦИИ

Цель работы: изучение методов считывания символической информации в программном пакете LabVIEW с использованием средств IMAQ Vision.

Теоретические сведения

QR-код (англ. *quick response* - быстрый отклик) - матричный код (двумерный штрих-код), разработанный и представленный японской компанией Denso-Wave в 1994 году.

В отличие от старого штрих-кода, который сканируют тонким

лучом, QR-код определяется датчиком или камерой смартфона как двумерное изображение. Три квадрата в углах изображения и меньшие синхронизирующие квадратики по всему коду позволяют нормализовать размер изображения и его ориентацию, а также угол, под которым датчик расположен к поверхности изображения. Точки переводятся в двоичные числа с проверкой по контрольной сумме.

Основное достоинство QR-кода - это легкое распознавание сканирующим оборудованием, что дает возможность использования в торговле, производстве, логистике.



Рис. 8.1 QR-код сообщения «Программируем на Lab VIEW»

Расознавание QR-кодов.

Для распознавания QR-кодов, надо получить цветное изображение и преобразовать его в изображение градаций серого.

Для нахождения QR-кода на изображении воспользуемся функцией, которая находится в палитре: Vision and Motion – Machine Vision – Instrument Readers – IMAQ Read QR Code.

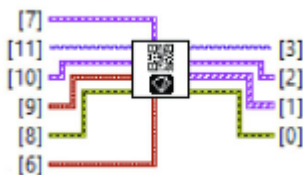


Рис. 8.2 Функция считывания QR-кода

Взять нужную информацию можно с выхода [2] QR Code Report, который находится в палитре на блок-диаграмме Function – Claster, Class – Unbundle By Name. Увеличение числа выходных параметров происходит с помощью функции Unbundle By Name. В соответствие этим полям необходимо поставить компоненты вывода -

Data и Bounding Box.

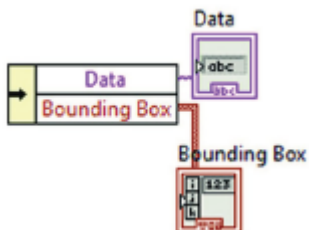


Рис. 8.3 Компоненты вывода

С помощью Indicator отобразим информацию. В поле Data будет отображаться закодированная информация, а в поле Bounding Box - координаты описывающего код прямоугольника. Их в дальнейшем можно использовать для нахождения местоположения самого объекта с RQ-кодом на изображении.

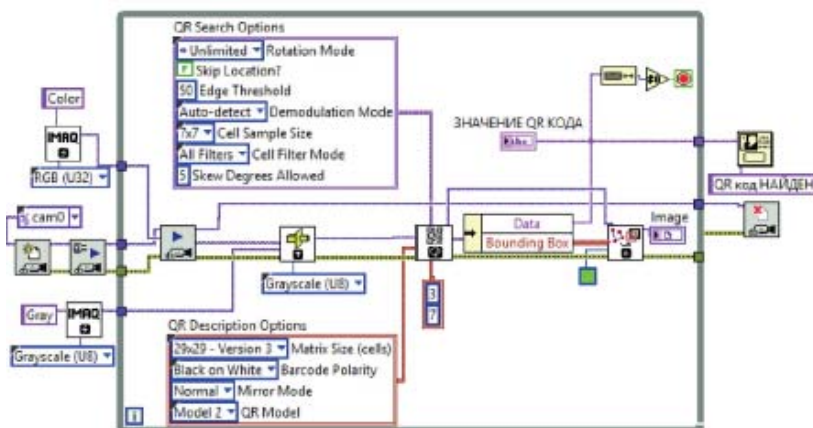


Рис. 8.4 Блок-диаграмма считывания QR-кода

Для фиксации момента распознавания QR-кода можно поставить диалоговое окно с диагностической информацией. Оно находится в палитре функций на блок-диаграмме Functions – Dialog&User Interface – One Button Dialog.



Рис. 8.5 Фиксация момента распознавания

Задание на практическую работу:

Изучить работу следующих функций IMAQ: Read LCD, Read Meter, Read Barcode. Собрать схему, демонстрирующую возможности каждой функции.

Отчет по практической работе должен содержать:

- 1) Скриншот и описание этапов выполнения;
- 2) Выводы, обобщающие полученные результаты.

РЕКОМЕНДУЕМЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Рабинович Е.В.* Методы и средства обработки сигналов: Учебное пособие / Новосибирск: Новосибирский государственный технический университет, 2009, 144 с.

2. *Рафаэл Гонсалес* Цифровая обработка изображений / Рафаэл Гонсалес, Ричард Вудс. М.: Техносфера, 2012, 1104 с.

3. *Тропченко А.А.* Методы вторичной обработки и распознавания изображений: Учебное пособие / Тропченко А.А., Тропченко А.Ю. СПб.: Университет ИТМО, 2015, 215 с.

4. *Тропченко А.Ю.* Цифровая обработка сигналов. Методы предварительной обработка: Учебное пособие / А.Ю. Тропченко, А.А. Тропченко. СПб.: Университет ИТМО, 2009, 88 с.

5. *Фисенко В.Т.* Компьютерная обработка и распознавание изображений: Учебное пособие / Фисенко В.Т., Фисенко Т.Ю. СПб.: Университет ИТМО, 2008, 195 с.

6. *Умняшкин С.В.* Основы теории цифровой обработки сигналов: Учебное пособие / Умняшкин С.В. М.: Техно-сфера, 2016, 528 с.

СОДЕРЖАНИЕ

Введение	3
Практическая работа № 1. Геометрические и алгебраические преобразования изображений	3
Практическая работа № 2. Форматы хранения, сжатия и передачи цифровых изображений и видео	6
Практическая работа № 3. Платформа NI Vision: захват изображений и видео в LabVIEW	13
Практическая работа 4. Методы обработки изображений: бинаризация и фильтрация полутоновых изображений	15
Практическая работа 5. Обработка цветных изображений	18
Практическая работа № 6. Методы анализа изображений: выделение и анализ связанных областей; контуров на полутоновых изображениях; выделение геометрических примитивов	20
Практическая работа № 7. Обнаружение объектов, заданных эталонами ..	23
Практическая работа № 8. Считывание символьной информации	25
Рекомендуемый библиографический список	29

МЕТОДЫ И АЛГОРИТМЫ ОБРАБОТКИ СИГНАЛОВ И ИЗОБРАЖЕНИЙ

*Методические указания к практическим работам
для студентов магистратуры направления 15.04.04*

Сост.: *А.В. Бойков, А.А. Виноградова*

Печатается с оригинал-макета, подготовленного кафедрой
автоматизации технологических процессов и производств

Ответственный за выпуск *А.В. Бойков*

Лицензия ИД № 06517 от 09.01.2002

Подписано к печати 27.06.2019. Формат 60×84/16.
Усл. печ. л. 1,7. Усл.кр.-отт. 1,7. Уч.-изд.л. 1,5. Тираж 50 экз. Заказ 632. С 232.

Санкт-Петербургский горный университет
РИЦ Санкт-Петербургского горного университета
Адрес университета и РИЦ: 199106 Санкт-Петербург, 21-я линия, 2