

# ОБРАБОТКА ВИДЕОСИГНАЛОВ ДЛЯ ИДЕНТИФИКАЦИИ ОБЪЕКТОВ НА ЖЕЛЕЗНОДОРОЖНОМ ПЕРЕЕЗДЕ

Н.Н. Васин, А.М. Баранов

Поволжская государственная академия телекоммуникаций и информатики

## Аннотация

Рассматривается метод обработки видеосигналов для распознавания объектов, которые могут появиться на железнодорожном переезде. Описываемые способы основываются на математических методах обработки изображений. Рассматривается способ избавления от шума при сравнении изображений.

## Постановка задачи

В общей структуре аварийности происшествия на железнодорожных переездах характеризуются значительными моральными и материальными потерями, вызывают широкий общественный резонанс. Особенно тяжелые последствия возникают при столкновениях автобусов с поездами. В последние годы наблюдается уменьшение числа переездов, обслуживаемых дежурным работником, следствием чего становится увеличение нерегулируемых переездов, что создает реальные предпосылки к увеличению аварийности [1].

Для увеличения безопасности нерегулируемых переездов разрабатывается система видеоконтроля за переездом. Такая система может строиться на базе видеокамеры, регистрирующей происходящее на переезде с того момента, когда переезд закрылся для автомобилей, и процессора.

Процессор, получая сигнал с видеокамеры, выясняет, есть ли какой-либо объект в зоне переезда. По мере приближения поезда к переезду, система продолжает оценивать ситуацию на переезде, и в случае необходимости передает сигнал на поезд о необходимости экстренного торможения или каких-либо других действий направленных на избежание или уменьшение последствий аварийной ситуации.

## Принцип функционирования системы

При создании такой системы необходимо решить ряд задач:

- определить присутствие постороннего объекта на переезде;
- идентифицировать объект;
- удалить шум с изображения.

Первую задачу можно решить несколькими способами. Существуют алгоритмы, позволяющие находить различия на двух изображениях методом вычитания одного изображения из другого. Этот способ предполагает попиксельное сравнение кадра номер  $n$  с кадром  $n + 1$  [2, 3]. Поскольку разрабатываемая система должна функционировать в реальном масштабе времени, то к ней предъявляются требования высокого быстродействия. В данной статье рассматривается метод обработки разностного изображения с повышенным быстродействием, т.к. важна скорость определения аварийной ситуации.

Упрощенная структурная схема системы, представлена на рис. 1.

В качестве О-процессора используется персональный компьютер (ПК). Программное обеспече-

ние, установленное на ПК, фиксирует опорный кадр, устанавливаемый, когда переезд пуст. После этого создается двумерный массив данных, куда заносится цвет каждого пикселя опорного изображения.

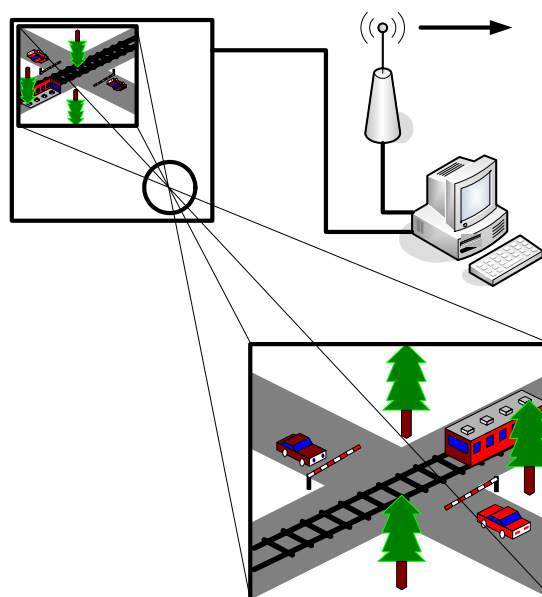


Рис. 1. Структурная схема системы

## Алгоритм функционирования системы

При закрытии переезда начинается функционирование системы. Получаемое с видеокамеры текущее изображение анализируется на предмет выявления постороннего объекта. Для этого сравниваются цвета пикселей нового изображения с цветами соответствующих пикселей опорного изображения. Сравнение производится не каждого пикселя, а только некоторых из группы, т.к. для сравнения всех пикселей быстродействие современных аппаратных средств не достаточно. Процесс выбора сравниваемых пикселей размером  $320 \times 240$  пикселей приведен на рис. 2. Данное изображение разбивается на 80 прямоугольников, т.е. получится 10 столбцов и 8 строк. В каждом из таких прямоугольников находится  $320 \times 240 / 80 = 960$  пикселей. Прямоугольник имеет размеры  $32 \times 30$ . После этого полученный прямоугольник разбивается еще на 80 более маленьких прямоугольников размером  $4 \times 3$  пикселя.

Первоначально производится сравнение только показанных на рисунке серых пикселей, т.е. из двенадцати пикселей будет проверен только один. Сле-

довательно, при сравнении двух совершенно одинаковых изображений будет опрошено только 8,3% всех пикселей. Такой алгоритм позволяет увеличить быстродействие системы примерно на порядок. На рис. 3 приведен пример сектора 4×3 пикселя.

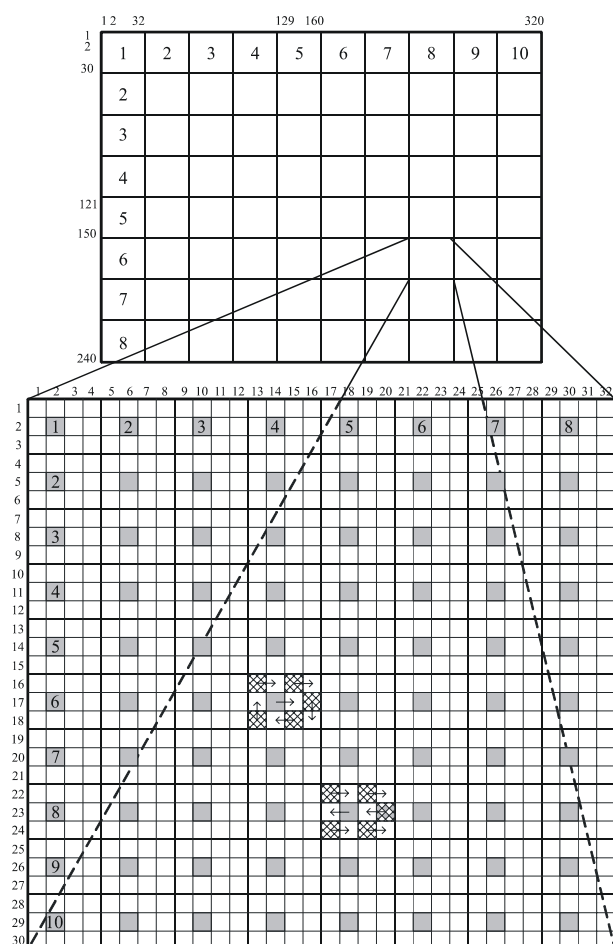


Рис. 2. Алгоритм разбиения изображения

$i-1$	$i$	$i+1$	$i+2$
$j-1$	$j-1$	$j-1$	$j-1$
$i-1$	$i$	$i+1$	<b><math>i+2</math></b>
$j$	$j$	$j$	<b><math>j</math></b>
$i-1$	$i$	$i+1$	$i+2$
$j+1$	$j+1$	$j+1$	$j+1$

Рис. 3. Схема сектора 4×3 пикселя

При сравнении двух изображений, вначале сравниваются только серые пиксели  $(i, j)$ . Если данные пиксели одинаковы по своим цветам, то рассмотрение в данном секторе прекращается и принимается, что одинаковы и остальные пиксели этого сектора. Если же пиксели  $(i, j)$  различны, то проверяются пиксели  $(i+2, j)$  (выделено жирным). Если и эти пиксели различны, то принимается, что пиксели с координатами  $(i-1, j)$  и  $(i+1, j)$  тоже различны. Далее про-

веряются пиксели  $(i-1, j-1)$ ,  $(i+1, j-1)$ ,  $(i-1, j+1)$ ,  $(i+1, j+1)$ . При различии цветов пикселей  $(i-1, j-1)$  принимается различие цветов и пикселей  $(i, j-1)$ ; при различии  $(i+1, j-1) - (i+2, j-1)$ ; при различии  $(i-1, j+1) - (i, j+1)$ ; при различии  $(i+1, j+1) - (i+2, j+1)$ . Таким образом, будет проверен весь сектор, и можно переходить к следующему.

При этом может получиться, что различными окажутся только некоторые пиксели. Так как при сравнении сегментов сравнивается 50% пикселей, то это говорит о некоторой погрешности, но эта погрешность, как показали эксперименты, незначительно сказывается на выделении объектов.

### Пример функционирования системы

Далее приведен пример функционирования системы. Опорное изображение  $n$ , зафиксированное с помощью видеокамеры, представлено на рис. 4.



Рис. 4. Опорное изображение  $n$

Изображение  $n+1$ , полученное через четыре секунды, представлено на рис. 5.



Рис. 5. Изображение  $n+1$

Результат сравнения каждого пикселя кадра  $n+1$  с соответствующим пикселем кадра  $n$  представлен на рис. 6. Главным недостатком данного метода яв-

ляется длительное время операции сравнения, что не приемлемо для разрабатываемой системы реального времени. Кроме того, при таком методе сравнения получается зашумленное изображение.

При обработке изображений описанным выше методом, когда сравнивается один из двенадцати пикселей, получен результат, приведенный на рис. 7.

Данное изображение зашумлено меньше, чем рис. 6. Качество выделяемого объекта (машины) при этом снизилось, но данное явление является компромиссом между качеством и скоростью обработки. Результат эксперимента показал, что обработка данным методом производится, примерно, в 5,3 раз быстрее сравнения методом каждого пикселя. Такое увеличение скорости обработки вполне оправдывает незначительные потери в качестве изображений.



Рис. 6. Разность кадров  $n+1$  и  $n$  с обработкой каждого пикселя

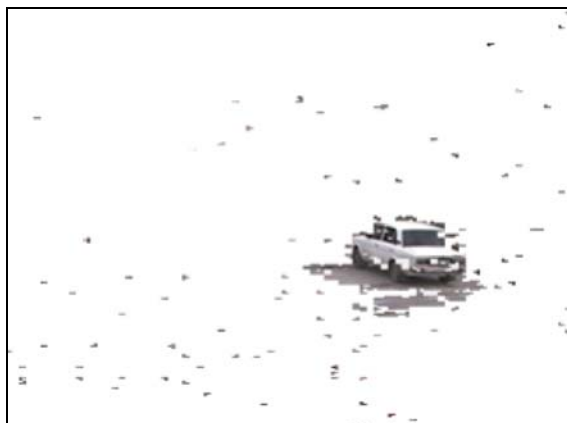


Рис. 7. Разность кадров  $n+1$  и  $n$  с обработкой группы пикселей

Эксперименты проводились на ПК с ЦП Athlon 64 3 GHz и оперативной памятью 512 MB. Было получено, что при сравнении методом каждого пикселя затрачивалось 0,69 секунды, при рассмотрении примера с одним объектом на переезде. А при сравнении, предложенным методом было затрачено 0,13 секунды. Если брать пример с несколькими объектами на переезде, то время обработки изобра-

жения увеличивалось, но при использовании предложенного алгоритма время преобразования не превышало 0,5 секунды. Такая скорость преобразования дает реальное представление о ситуации, происходящей на переезде.

Следующим шагом обработки изображения является избавление от шума, или, хотя бы, его уменьшение. Для решения этой задачи предлагается разбить исходное изображение на 80 сегментов размером  $32 \times 30$  пикселей. Теперь можно, обработав один сегмент, сказать, является ли его содержимое шумом или нет. Так в одном сегменте 960 пикселей. Установив порог шума в  $m$  пикселей можно контролировать шум. Если в сегменте различных пикселей меньше  $m$ , то рассматриваемый сегмент является шумом. Следовательно, изменяя порог шума можно добиться вполне приемлемого результата. На рис. 8 приведен результат сравнения кадра  $n+1$  с кадром  $n$  и порогом шума  $m=16$ . Если из 960 пикселей различаются не более 16, то эти пиксели являются шумом и дальнейшему рассмотрению не подлежат.



Рис. 8. Результат подавления шума

Еще одним важным моментом обработки видеосигнала является определение цвета пикселя и его сравнение с цветом пикселя другого изображения. В создаваемой системе используется RGB-модель представления цвета, когда каждая составляющая может быть представлена 256 градациями. Следовательно, рассматриваются 24-битные изображения. Получив цвет пикселя опорного изображения его необходимо разложить на три составляющие. После этого нужно сформировать интервалы по трем составляющим. Если составляющая красного цвета равна  $R$ , то интервал будет  $[R-k; R+k]$ , где  $k$  – некоторое положительное число. Значение  $k$  выбирается из соображений достижения определенной точности сравнения пикселей. И если значение составляющей красного цвета пикселя второй картинки попадает в сформированный интервал, то организуются интервалы по двум оставшимся составляющим и производится подобная проверка. Если хотя бы одно значение составляющей не попало в соответствующий ей интервал, то выносится решение о различии цветов данных пикселей. Иначе пиксели одинаковые.

### *Заключение*

1. Применение предложенного метода приводит к увеличению скорости обработки изображений при сохранении точности нахождения объектов.

2. Данный метод позволяет избавляться от шума на стадии сравнения изображений, а не на последующих стадиях, что облегчает дальнейшую обработку получаемой информации.

3. Неоспоримым достоинством рассмотренного метода является не строгое сравнение изображений. Например, если из-за облачности изменится освещенность переезда, то система отреагирует на такое изменение вполне адекватно и не прекратит свое

функционирование из-за предусмотренного интервала цветности по всем трем составляющим цвета.

### *Литература*

1. Чикин В.Н., Поздняков В.А., Тюпкин Ю.А. Железнодорожные переезды: что показывает статистика // Железнодорожный транспорт, 2002. № 3.
2. Методы компьютерной обработки изображений // Под ред. В.А. Сойфера. М.: Физмалит, 2004.
3. Еремин С.Н., Малыгин Л.Л., Михайлов А.Е., Царев В.А. Опыт использования технологии обработки изображений при проектировании интеллектуальных транспортных систем // Искусственный интеллект в XXI веке: Труды международного конгресса ICAI 2001. М.: Физмалит, 2001. Т. 2.