

Таким образом был разработан и проанализирован алгоритм сжатия графики на основе искусственных нейронных сетей. Для создания соответствия между входным и выходным пространствами была применена Самоорганизующаяся карта Кохонена. Результаты моделирования данного алгоритма свидетельствуют о возможности практического применения данного вида ИНС для сжатия изображений.

Список использованных источников:

1. Бодянский, Е.В. Искусственные нейронные сети: архитектуры, обучение, применения. / Е.В. Бодянский, О.Г. Руденко – Харьков 2004. – 369 с.
2. Amerijckx, Christophe. Image Compression by Self-Organized Kohonen Map / С. Amerijckx, M. Verleysen, P. Thissen, J-D. Legat // IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 9, NO. 3 – MAY 1998 – P. 503-507.

МЕТОДЫ ФОРМИРОВАНИЯ ГИБКОГО ПЛАНА РАБОТ ПО КОНТРОЛЮ КАЧЕСТВА РАЗРАБОТКИ ПС

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

Новицкий А. А.

Насуро Е. В. – к.т.н. доцент

Повсеместное внедрение программных продуктов и рост сложности используемых технологий и ПС становится причиной роста цены ошибки в продукте, поэтому не теряет актуальности вопрос дальнейшей оптимизации процессов контроля качества (КК) разработки программных средств (ПС). Так как работы по контролю качества разработки востребованы с определения требований к продукту и до самого выпуска, выбор оптимальных для проекта работ по КК отвечает целям повышения надежности итогового продукта, уменьшения издержек, повышения рентабельности процесса разработки, и является актуальной темой для настоящей работы.

Время и бюджет, требуемые на планирование работ и непосредственно сами работы по КК на проекте тем больше, чем сложнее структура и функционал и выше требования к надежности разрабатываемого продукта. Необходимо учитывать и работы по проверке стрессоустойчивости системы, требующие специфической квалификации.

Учитывая многообразие классификаций видов тестирования, в данной работе рассматривается разделение по необходимости выполнения программного кода. Это позволяет четко выделить как минимум одну важную группу работ. По данному критерию методы тестирования делятся на статические, не требующие запуска программного кода, и динамические, непосредственно взаимодействующие с функционирующим приложением или его моделью.

Статические виды тестирования включают работы по верификации требований и анализ программного и скомпилированного кода, вследствие чего могут обеспечивать наиболее раннее выявление различных проблем, потому обязательны к включению в план. [1] Динамические методы тестирования в основном обеспечивают обнаружение дефектов функциональности в ходе проверки тестовых сценариев, или же отвечают задачам проверки устойчивости системы к превышению пределов нормального функционирования, закрытости от влияния извне. Подходы к динамическому тестированию наиболее разнообразны, что создает дополнительные трудности в выборе работ для полного неизбыточного покрытия потребностей разрабатываемого ПС. При этом, различные подходы по различным методам могут быть задействованы на разных этапах жизненного цикла ПС – прямое влияние на это оказывает наращивание функциональности, требующее как полноценной проверки новых фрагментов, так и контроля уже проверенных на предмет регрессии.

Разница в механизме доступа обуславливает специфику проведения тестирования, предпочтительного для каждого из методов. Так как условно-автоматизированное тестирование зачастую сокращает лишь механическую работу тестировщика и может не требовать глубокой экспертизы в сфере автоматизации процесса тестирования, в контексте исследуемой темы допустимо включить данный подход в категорию ручного тестирования. Таким образом, рассматривать следует критерии эффективности ручного и автоматизированного тестирования.

Независимо от того, будет ли реализовано динамическое тестирование посредством ручного или автоматизированного подхода, общим вариантом критерия оценки покрытия является покрытие, основанное на спецификации или требованиях. Главное требование состоит в покрытии некоторого утвержденного минимума – набора требований.

При том, что механизм планирования и оценки покрытия, разрабатываемого ПС тестированием может быть выбран в соответствии с потребностями системы и имеющейся документации, выбор применяемых методов и техник строго не регламентируется. План, методы и техники могут определяться командой на этапе планирования исходя из субъективных мнений, предпочтений, доступных ресурсов. [3] Таким образом, задача формирования полного неизбыточного плана работ по КК разработки, пригодного для внедрения в проекты,

процессы которых выстроены с использованием популярных моделей и методологий разработке на базе некоторых систем планирования процессов, должна быть рассмотрена с аналитической точки зрения.

Использование техник статистического тестирования должно позволить создать репрезентативный сценарий тестирования. В качестве некоторого критерия эффективности можно принять достижение надежностью однократного действия некоторого приемлемого для частного проекта случая. [4]

На основе полученного сценария и располагая некоторой статистикой с места планируемого внедрения плана работ, возможно более предметно рассчитать рентабельность автоматизации работ в процессе тестирования с учётом планируемой к использованию модели/методологии разработки с точки зрения затрат финансов и времени. Должны быть учтены возможности используемой системы планирования проектов с частным случаем жизненного цикла проекта как площадки для реализации выстраиваемого процесса. [2]

Список использованных источников:

1. Winston W. Royce. Managing the Development of Large Software Systems / Winston W. Royce // Article / Proceedings of IEEE WESCON 26: Article / The Institute of Electrical and Electronics Engineers, Inc. – 1970.
2. Extreme Programming Explained: Embrace Change, 2nd Edition: Book / Addison-Wesley. – NY, 2004.
3. Липаев В.В. Человеческие факторы в программной инженерии: рекомендации и требования к профессиональной квалификации специалистов: Учебник / Липаев В.В. – М.: Синтег, 2009. - 328 с.
4. Kirk Sayre. Improved Techniques for Software Testing Based on Markov Chain Usage Models: Dissertation / Kirk Sayre. – Knoxville: The University of Tennessee, 1999. –128с.

АНАЛИЗ АЛГОРИТМОВ ОБНАРУЖЕНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ

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

Русакович П.Л.

Лукашевич М. М. – к.т.н., доцент

При визуальном наблюдении за объектами, например, людьми или автомобилями, видео обрабатывается, чтобы охарактеризовать интересующие нас события посредством обнаружения движущихся объектов в каждом кадре. Большинство ошибок в задачах более высокого уровня, таких как отслеживание, часто возникают из-за ложного обнаружения. Поэтому, важно быстро и точно определить объект, для дальнейшего его отслеживания. Наиболее используемыми методами обнаружения являются метод межкадровой разности, метод вычитания фона и метод оптического потока.

Межкадровая разность

Обнаружение объектов при помощи разницы кадров относится к самому популярному методу первичного обнаружения движения, в результате которого, уже можно сказать, имеется ли в данном видеоряде движение. Но перед этим, кадры необходимо предварительно обработать для вычисления межкадровой разности. Алгоритм вычисления, в качестве исходных данных которого будет использоваться цветное видео в формате RGB, будет состоять из 4 шагов:

1. На вход поступают два кадра видеоряда, которые представляют собой две последовательности байт в формате RGB.
2. Производится попиксельное вычитание межкадровых разностей.
3. Для каждого пикселя вычисляется усредненное значение между исходными данными значений трёх компонент цвета.
4. Полученное значение сравнивают с заданным порогом. В конечном результате, выводится двоичная маска.

Таким образом, на выходе имеется сформировавшаяся двоичная маска. Каждый элемент представляет собой три компоненты цвета соответствующего пикселя исходных двух изображений из видеопотока. Единицы в маске помечаются в областях, где, возможно, происходит движение, но на данном этапе существует вероятность ложного срабатывания определенных элементов битовой маски, которые ошибочно могут быть установлены в 1. Для входных параметров можно использовать два последовательных кадра из видеопотока, однако при использовании кадров с большим интервалом, например, равным 1-3 кадра чувствительность алгоритма к мало подвижным объектам, так как они производят крайне малый сдвиг за один кадр и могут определяться, как шумовая составляющая кадра.

Недостатком такого подхода относится фиксация шума аппаратуры при сохранении данных с видеоряда. Шум тяжело свести к предельно допустимым значениям, поэтому с ним приходится бороться отдельно.

Достоинством алгоритма является его простота и малые требования к вычислительным ресурсам. Он широко использовался ранее так как, в распоряжении разработчиков отсутствовали необходимые вычислительные мощности.