

УДК 004.932.2

**АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ ОТОБРАЖЕНИЯ ТРЕХМЕРНЫХ  
ОБЪЕКТОВ НА ОСНОВЕ АНАЛИЗА ИЗОБРАЖЕНИЙ****Масалимов Тимур Рустемович,**ФГБОУ ВО «Уфимский Государственный Нефтяной Технический Университет», магистр  
второго курса.

mstimur@gmail.com

**Аннотация**

В данной статье рассмотрен способ оценки качества изображений на основе комплексного применения методов полного сравнения изображений и индекса структурного сходства. На основе описанного способа предложен метод автоматизации проверки отображения трехмерных сцен и объектов для веб и настольных приложений.

**Ключевые слова:** Визуальное тестирование, SSIM, индекс структурного сходства, анализ изображений, оценка качества изображения, сравнение изображений.

**AUTOMATION OF TESTING THE DISPLAY OF THREE-DIMENSIONAL  
OBJECTS BASED ON IMAGE ANALYSIS****Timur R. Masalimov**

Ufa State Petroleum Technological University, master of the 2nd year.

mstimur@gmail.com

**ABSTRACT**

This article discusses a technique for assessing the quality of images based on the complex application of methods for a complete comparison of images and an index of structural similarity. Based on the described technique, a method for automating the verification of the display of three-dimensional scenes and objects for web and desktop applications is proposed.

**Keywords:** Visual testing, SSIM, structure similarity index, image analysis, image quality assessment, image comparison.

**Введение.** Автоматизированное тестирование на сегодняшний день активно применяется в процессах разработки программного обеспечения (далее ПО) и продолжает успешно развиваться. Автотесты позволяют избавить часть команды разработки от повторяющихся рутинных проверок, освобождая время на другие более приоритетные

задачи. Грамотное применение автоматизированных тестов может снизить количество дефектов, попавших на продуктивную среду, уменьшая производственные и финансовые затраты на разработку и поддержку программы.

Одним из типов автоматизированного тестирования является визуальное тестирование. Это способ проверки визуального отображения какой-то части графического представления ПО на предмет соответствия ожидаемому результату. Основной задачей визуального тестирования является обнаружение дефектов в отображении элементов графического интерфейса. В общем случае визуальное тестирование – это способ сравнить реальное изображение, представленное на экране, в том виде каким его видит пользователь, с ранее сохраненным шаблоном (эталонном). Большинство существующих инструментов визуального тестирования работают по следующему принципу: при первом запуске тестов изображения сохраняются как эталонные. При следующих запусках тестов скриншоты сравниваются с соответствующими эталонными изображениями и сообщается об обнаруженных различиях [1]. Такие инструменты в основном предназначены для проверки верстки веб-страниц.

**Цель исследования.** В данной работе предложен способ автоматизированной визуальной проверки отображения объектов на трехмерной сцене. В качестве примера представлено простое веб-приложение, отображающее трехмерную сцену с возможностью интерактивного взаимодействия с ней. Доступ пользователя к приложению осуществляется через браузер. Цель работы заключается в поиске способа автоматизированной проверки корректности отображения элементов трехмерных сцен как веб-решений, так и настольных приложений.

**Материалы и методы исследования.** В основе представленного метода проверки лежит сравнение изображений трехмерной сцены или ее отдельных ключевых элементов. Для решения проблемы подобным методом подходит простой алгоритм полного (попиксельного) сравнения тестируемого изображения с эталонным. Но этот метод имеет значимый недостаток – небольшие изменения в изображении обязательно будут обнаружены, даже если они в действительности не являются дефектом. Обнаруженные изменения будут указывать на то, что тест провален, так как существуют различия в отображении трехмерной сцены. Поэтому существует необходимость доработки описанного алгоритма с целью сделать проверку менее чувствительной для несущественных различий, таких как сглаживание, небольшие шумы, минимальная разница в оттенках цветов и т.п. Для этого предложен способ комплексного применения полного сравнения совместно с SSIM. SSIM (structure similarity index) – это показатель полной оценки качества эталонного изображения (FR-IQA), основанный на теории о том, что HVS хорошо приспособлен для извлечения структурной информации из сцен [2]. HSV (Human Visual System) – модель, применяемая в обработке изображений, описывающая принципы того, как человек визуально воспринимает окружающий его мир с биологической и психологической точек зрения [3].

В качестве объекта проверки в данной работе выступает веб-страница с трехмерной сценой, на которой расположено несколько видов геометрических объектов (в т.ч. плоскость с наложенной текстурой). Объекты на сцене статичны. Сцена создана с помощью Java Script фреймворка Babylon.js. Пользователь может взаимодействовать со сценой, управляя камерой, вращая ее и меняя масштаб. Предположим, что при тестировании приложения нужно убедиться, что после каждого его обновления все элементы на сцене отображаются корректно. Эталонный вид 3D сцены представлен на рисунке 1.

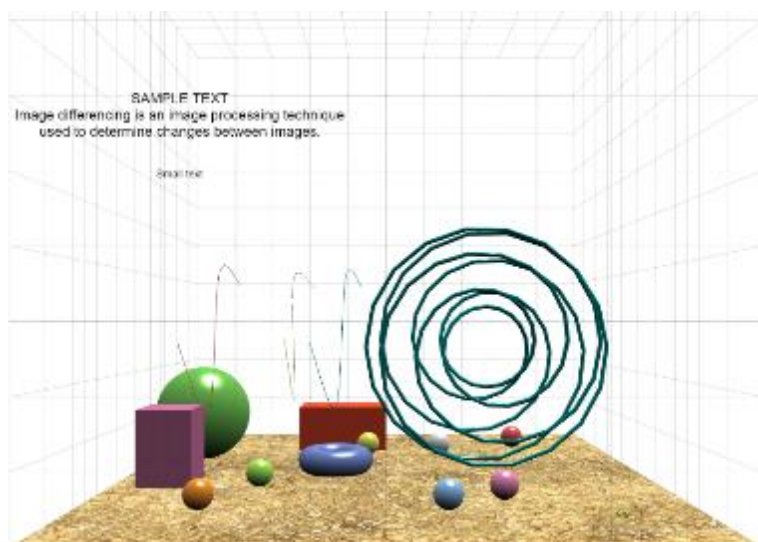


Рисунок 1 – Вид трехмерной сцены тестируемого приложения

Необходимо провести следующие проверки:

- убедиться, что все объекты присутствуют на сцене;
- убедиться, что цвета объектов соответствуют эталону;
- убедиться, что на сцене отсутствуют лишние объекты и дефекты рендера, ошибки в отображении текстур, шейдеров, освещения;

Методология предполагает создание изображений тестируемой сцены с нескольких ракурсов (спереди, сзади, справа, слева и сверху). На веб-страницах процесс захвата скриншотов легко автоматизировать, используя инструмент Selenium (подобные инструменты существуют и для настольных приложений). Важно, чтобы изображения каждый раз создавались под одинаковым ракурсом (углом наклона), чтобы при сравнении положение одних и тех же пикселей совпадало. Предварительно таким же образом создается группа эталонных изображений. Проверка заключается в попарном сравнении группы сделанных в ходе теста изображений с группой эталонных (рис. 2).

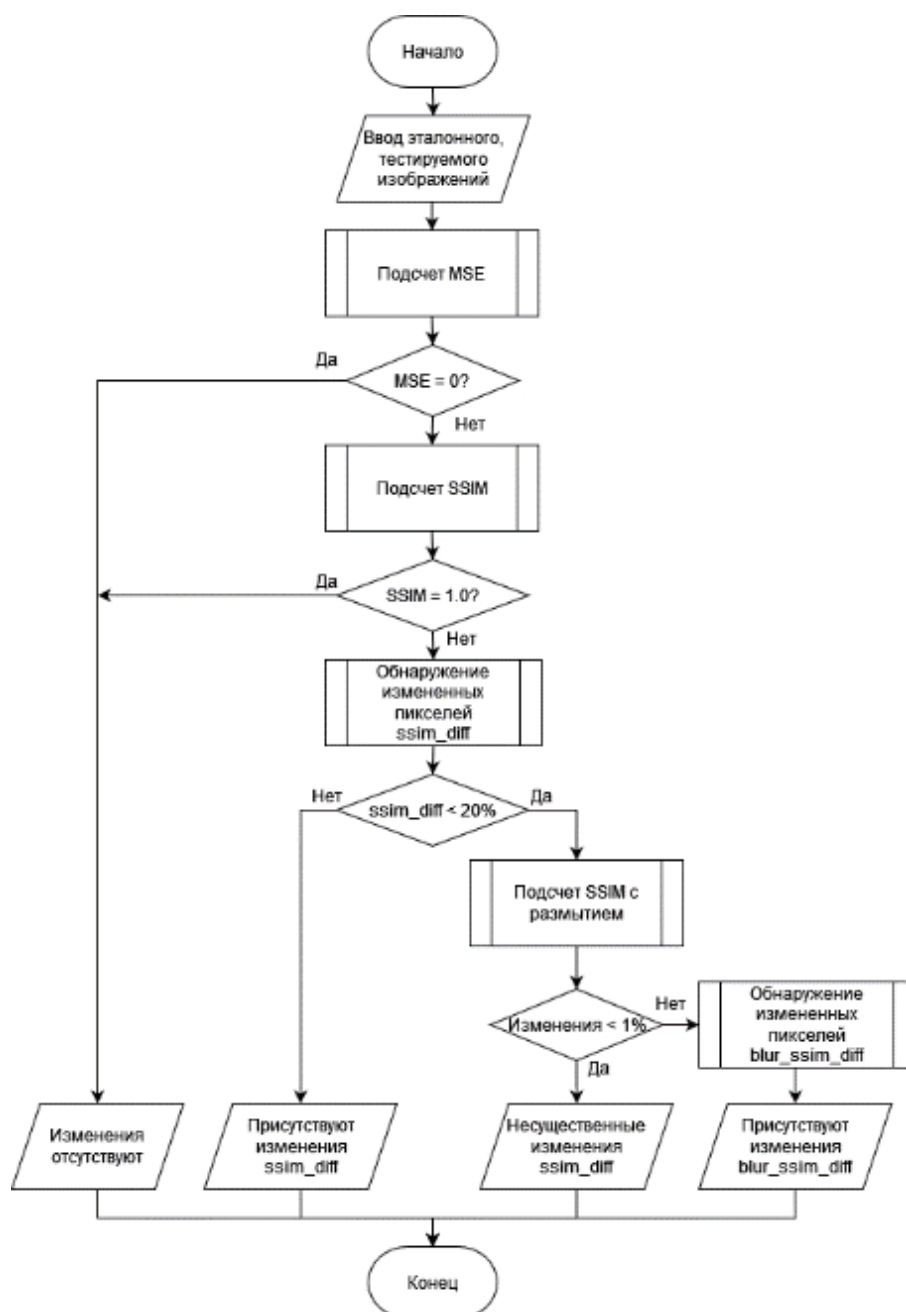


Рисунок 2 – Блок-схема процесса сравнения изображений

Первый этап проверки заключается в полном (попиксельном) сравнении изображений – для каждого пикселя вычисляется модуль разницы его значений на тестируемом и эталонном изображении. В качестве метрики используется среднеквадратичная ошибка (MSE). MSE для одного цветового канала (где N и M – высота и ширина изображения) [4]:

$$MSE = \frac{1}{NM} \sum_{i=1}^N \sum_{j=1}^M (Y_{ij} - X_{ij})^2$$

MSE с учетом трех цветовых каналов (RGB):

$$MSE = \frac{1}{NM} \sum_{c=1}^3 \sum_{i=1}^N \sum_{j=1}^M (Y_{cij} - X_{cij})^2$$

Если значение  $MSE = 0$ , то изображения полностью идентичны и тест можно считать пройденным.

Если  $MSE \neq 0$ , то применяется сравнение на основе алгоритма структурного сходства SSIM [5]:

$$S_1(x, y) = l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$S_2(x, y) = c(x, y)s(x, y) = \frac{2s_{x,y} + c_2}{s_x^2 + s_y^2 + c_2}$$

$$SSIM(x, y) = S_1(x, y)S_2(x, y) = \frac{(2\mu_x\mu_y + c_1)(2s_{x,y} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(s_x^2 + s_y^2 + c_2)}$$

где  $\mu_x$  и  $\mu_y$  – средние значения выборки пикселей  $x$ ,  $y$ ,  $s_x^2$  и  $s_y^2$  – дисперсии  $x$ ,  $y$ ,  $s_{x,y}$  – ковариация  $x$  и  $y$ .

SSIM состоит из трех индексов: индекса яркости ( $l$  – luminance), контрастности ( $c$  – contrast) и структуры ( $s$  – structure). Контраст отражает разницу в яркости или степень изменения яркости. Структура отражает корреляцию Пирсона яркости между двумя изображениями [6].

Первый этап проверки нацелен на обнаружение общих изменений со стандартными параметрами SSIM. Определяется граница чувствительности к изменениям (в данном примере принято 20%). Если число пикселей с изменениями составляет менее заданной границы, то выполняется второй этап проверки. Он выполняет поиск изменений с использованием SSIM, игнорируя небольшие области размытия, шумы и незначительные изменения оттенков. На этом этапе перед SSIM происходит размытие всего изображения (как эталонного, так и тестируемого). Также немного снижается чувствительность SSIM к тем частям изображения, разница пикселей которых составляет не больше 10-20 единиц. Если вторая проверка SSIM не выявила изменений, то тест считается пройденным, и результаты первой проверки можно интерпретировать как незначительные различия, не влияющие на отображение сцены.

Недостаток представленной методологии связан с тем, что SSIM имеет слабую чувствительность к небольшим изменениям в оттенках и контрасте, т.к. основным назначением алгоритма является выделение информации о структуре объектов и сцене в целом. Поэтому, если небольшие изменения оттенков цветов на сцене критически важны для проверки, то второй этап сравнения SSIM рекомендуется использовать с осторожностью. Например, в итоговом отчете о тестировании прилагать результаты первого этапа сравнения SSIM с указанием на то, что обнаруженные различия могут быть значимыми при сравнении цветов.

**Результаты.** Результаты работы описанных трех алгоритмов представлены в таблице 2 в виде изображений. Каждое изображение было изменено определенным способом (способы описаны в таблице 1). Измененные изображения сравнивались с эталонным (рис. 1). Метрики сравнения для каждого алгоритма также представлены в таблице 1.

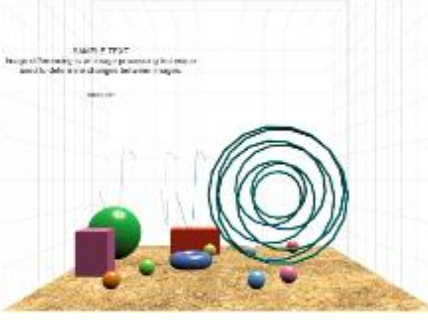
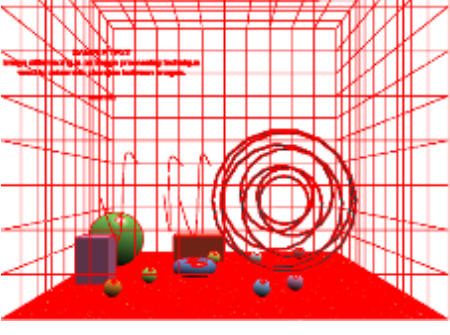
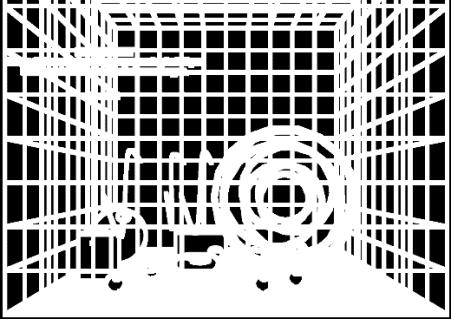

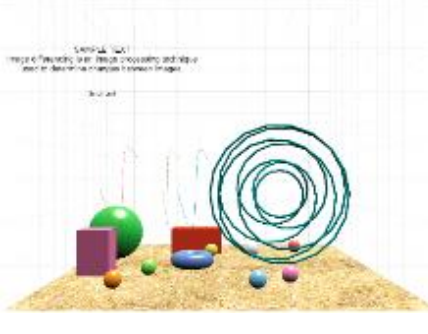
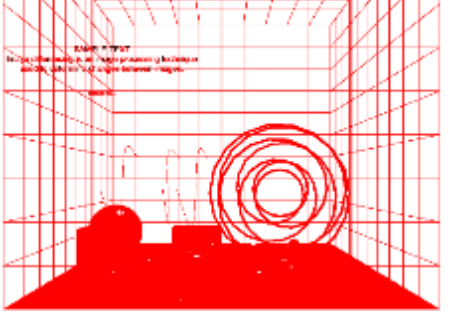
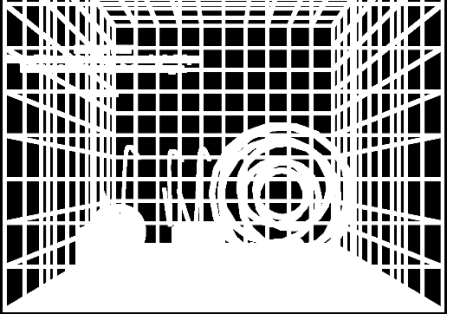
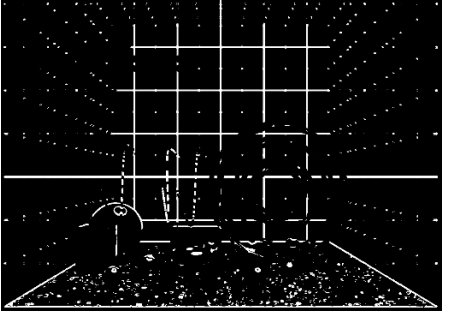
Для MSE изменения в изображениях подсвечены красным цветом, для SSIM – белым.

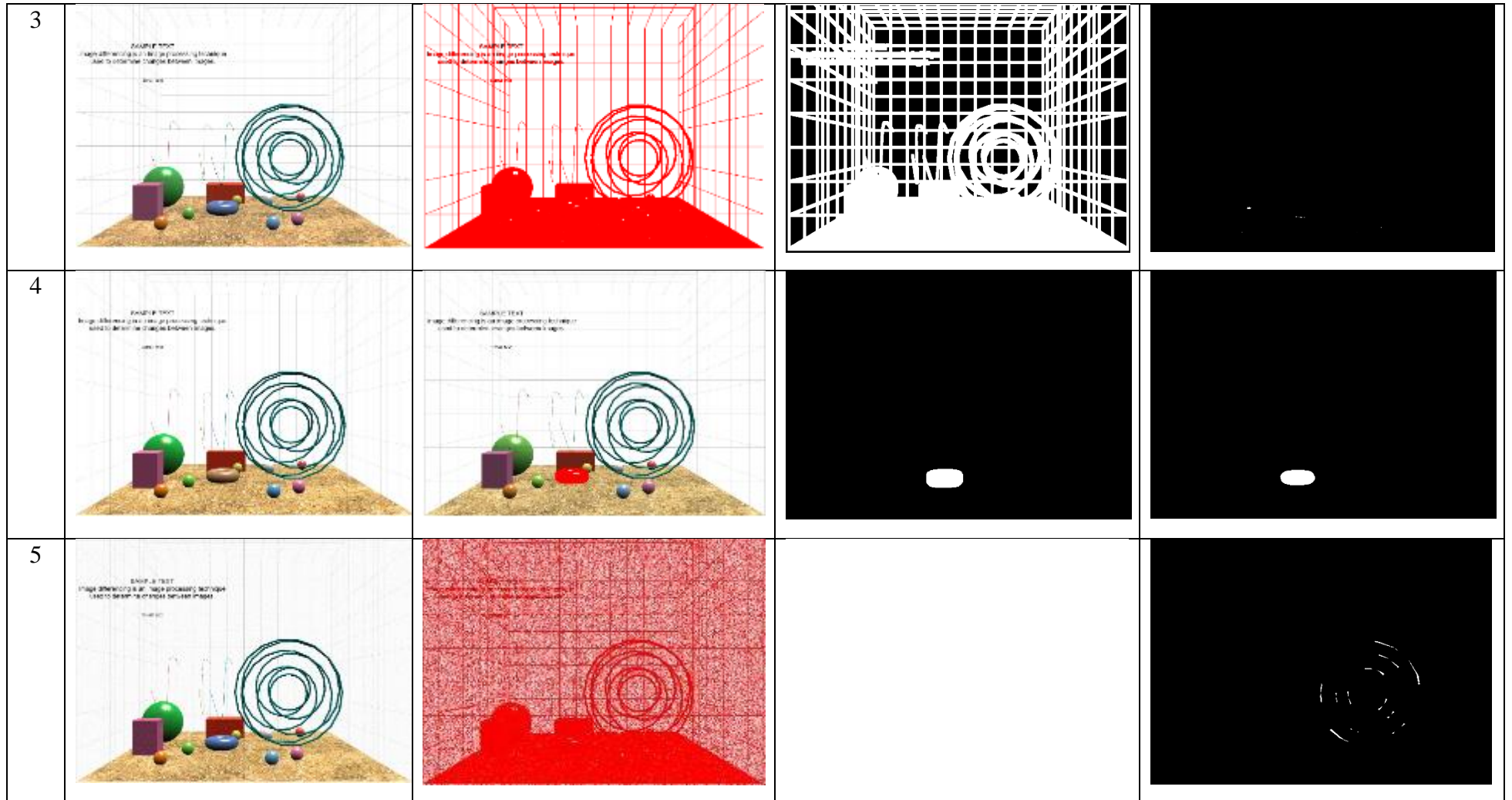
Таблица 1 – Результаты сравнения изображений с эталоном (метрики)

№	Изменения	MSE	SSIM	SSIM с размытием
1	Малое размытие по всему изображению	36.034	0.98227	0.99734
2	Повышенная яркость	49.092	0.98536	0.99248
3	Пониженная контрастность	37.695	0.99708	0.99832
4	Существенно изменен цвет объекта (тор)	1.008	0.99955	0.99958

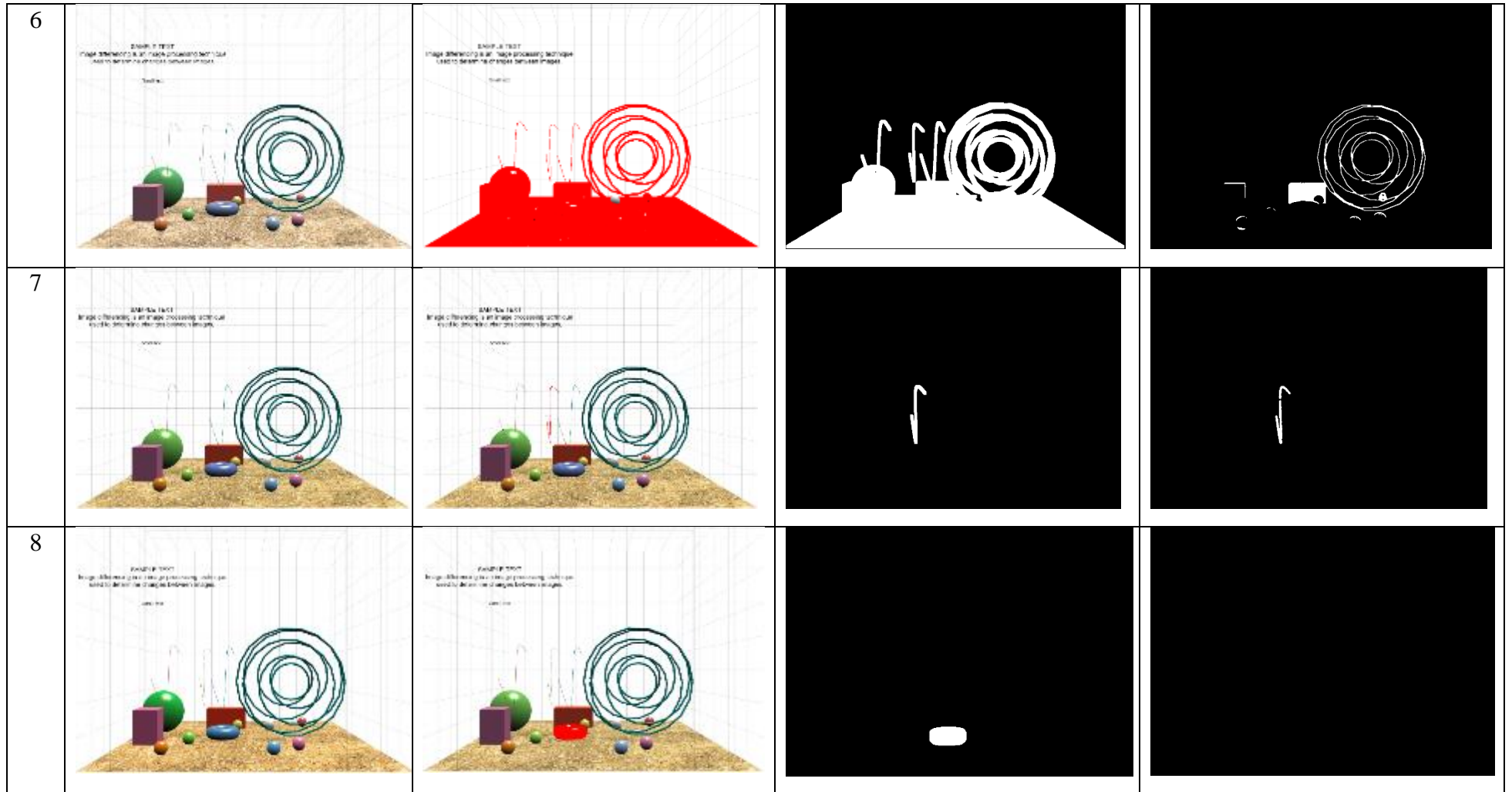
5	Добавлен шум	15.022	0.99516	0.99929
6	Пониженная насыщенность	20.847	0.99615	0.99672
7	Удаление небольшого объекта (кривой)	0.091	0.99921	0.99954
8	Незначительно изменен цвет объекта (тор)	0.273	0.99998	0.99999
9	Незначительно размыт текст	2.578	0.99945	0.99986

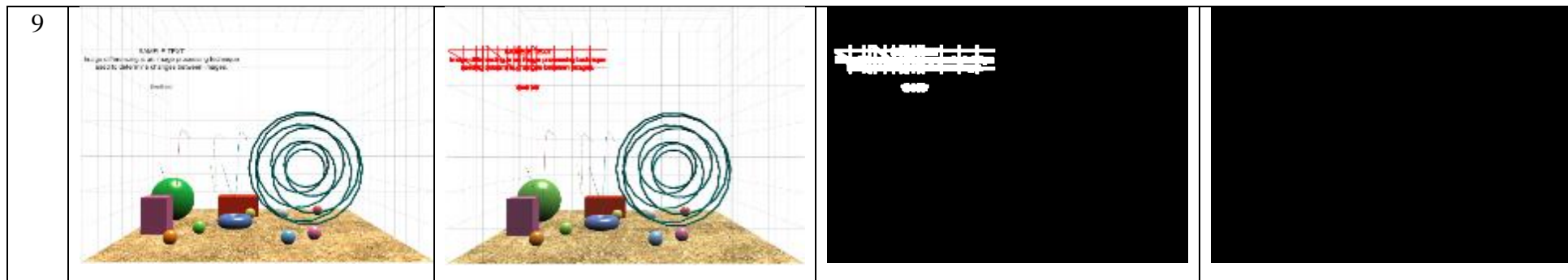
Таблица 2 – Результаты сравнения изображений с эталоном (изображения с обнаруженными изменениями)

№	Изображение	MSE	SSIM	SSIM с размытием
1				
2				









**Заключение.** На основе полученных результатов можно сделать вывод о том, что представленная методика, состоящая из полного (попиксельного) сравнения, простого SSIM и SSIM с предварительным размытием, подходит для сравнения изображений, в которых допустимы очень незначительные изменения. При этом алгоритм полного сравнения будет обнаруживать подобные изменения, что делает затруднительным его применение в чистом виде из-за риска ложных срабатываний. Дополнительное применение SSIM с предварительным размытием позволяет решить эту проблему, делая сравнение менее чувствительным к незначительным изменениям.

Тем не менее, следует учитывать слабую чувствительность SSIM к небольшим изменениям оттенков цветов. В традиционных методах оценки качества изображения, к которым можно отнести SSIM, оценка цветного изображения всегда выполняется путем оценки яркости или его преобразования в градации серого, поэтому информация о цвете в изображении в значительной степени игнорируется, и это влияет на точность результата [7]. В эксперименте под номером 13 (таблица 2) SSIM с размытием не выявил никаких дефектов, хотя в действительности объекты в тестируемом и эталонном изображениях имеют разные цвета. При внедрении методики в процесс автотестирования для таких случаев рекомендуется предусмотреть вывод предупреждения о нестабильности результата и необходимости дополнительной ручной проверки со стороны тестировщика в связи с наличием незначительных дефектов.

Несмотря на это, описанный метод совместного использования SSIM с алгоритмом полного сравнения применим для проверки отображения трехмерных объектов и сцены в процессах автоматизированного тестирования при разработке ПО.

#### **Список литературы:**

1. Axelrod A. Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects / Axelrod A., Israel M. – N.Y.: Apress, 2018. – 529 p.
2. Structural similarity index (SSIM) revisited: A data-driven approach / Illya Bakurov [and others] // Expert Systems with Applications. – 2022. – No. 189(11).
3. Umme S. Image quality assessment and human visual system / Xinbo Gao [and others] // Visual Communications and Image Processing. – 2010. – No. 7744.
4. Image Quality Assessment through FSIM, SSIM, MSE and PSNR-A Comparative Study / Umme Sara, Morium Akter, Mohammad Shorif Uddin // Journal of Computer and Communications. – 2019. – No. 7(3). – pp. 8-18.
5. Applicability of Existing Objective Metrics of Perceptual Quality for Adaptive Video Streaming / Jacob Sjøgaard [and others] // Electronic Imaging. – 2016. – No. 13.
6. Implementation of the Structural Similarity (SSIM) Index as a Quantitative Evaluation Tool for Dose Distribution Error Detection / Jiayuan Peng [and others] // Medical Physics. – 2020. – No. 47(4).
7. Mohammed H. H. Structural Similarity Measure for Color Images / Mohammed Hassan Husain, Chakravarthy Bhagvati // International Journal of Computer Applications. – 2012. – No. 43(14). – pp. 7-12.

**References:**

1. Axelrod A. Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects / Axelrod A., Israel M. – N.Y.: Apress, 2018. – 529 p.
2. Structural similarity index (SSIM) revisited: A data-driven approach / Illya Bakurov [and others] // Expert Systems with Applications. – 2022. – No. 189(11).
3. Umme S. Image quality assessment and human visual system / Xinbo Gao [and others] // Visual Communications and Image Processing. – 2010. – No. 7744.
4. Image Quality Assessment through FSIM, SSIM, MSE and PSNR-A Comparative Study / Umme Sara, Morium Akter, Mohammad Shorif Uddin // Journal of Computer and Communications. – 2019. – No. 7(3). – pp. 8-18.
5. Applicability of Existing Objective Metrics of Perceptual Quality for Adaptive Video Streaming / Jacob Søggaard [and others] // Electronic Imaging. – 2016. – No. 13.
6. Implementation of the Structural Similarity (SSIM) Index as a Quantitative Evaluation Tool for Dose Distribution Error Detection / Jiayuan Peng [and others] // Medical Physics. – 2020. – No 47(4).
7. Mohammed H. H. Structural Similarity Measure for Color Images / Mohammed Hassan Husain, Chakravarthy Bhagvati // International Journal of Computer Applications. – 2012. – No. 43(14). – pp. 7-12.