

К.В. САЗОНОВ, М.В. ТАТАРКА, А.С. ЦЫГАНОВ, В.Е. БЕССОЛЬЦЕВ
**ИДЕНТИФИКАЦИЯ АБОНЕНТСКИХ ТЕРМИНАЛОВ
ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЕЙ
НА ОСНОВЕ МОДЕЛИ ФОРМИРОВАНИЯ ИЗОБРАЖЕНИЙ В
СОВРЕМЕННЫХ КОМПЬЮТЕРНЫХ СИСТЕМАХ**

Сазонов К.В., Татарка М.В., Цыганов А.С., Бессольцев В.Е. Идентификация абонентских терминалов информационно-телекоммуникационных сетей на основе модели формирования изображений в современных компьютерных системах.

Аннотация. Одной из важных задач таких теорий, как теория распознавания образов и теория защиты информации, является задача идентификации абонентских терминалов информационно-телекоммуникационных сетей.

Актуальность данной темы обусловлена необходимостью исследования методов идентификации абонентских терминалов влияющих на построение систем защиты информации транслируемой в информационно-телекоммуникационных сетях.

Основными параметрами, позволяющими уникально идентифицировать абонентские терминалы в сети, являются адресно-коммутационная информация, а также параметры, которые характеризуют программно-аппаратную часть компьютерной системы. На основе этих параметров формируются цифровые отпечатки абонентских терминалов.

Применение пользователями абонентских терминалов анонимных сетей и блокирования методов формирования и сбора параметров цифрового отпечатка не позволяет в ряде случаев достичь требуемой степени достоверности идентификации.

Из-за особенностей формирования цифровых изображений в современных компьютерных системах множество параметров преобразований накладывают свой отпечаток на выходной графический примитив, формируя тем самым цифровой отпечаток абонентского терминала, который зависит от размещения семплов в пикселе, от используемых алгоритмов вычисления степени влияния пикселей, а также от применяемых процедур сглаживания изображений в графической подсистеме.

Предлагается оригинальная модель формирования изображений средствами веб-обозревателя абонентского терминала, позволяющая повысить степень достоверности идентификации в условиях анонимизации пользователей информационно-телекоммуникационных сетей.

Выявлены особенности формирования цифровых изображений в графических подсистемах современных компьютерных систем. Данные особенности позволяют осуществлять идентификацию в условиях априорной неопределенности относительно режимов и параметров передачи информации.

Ключевые слова: идентификация, модель формирования изображений, абонентский терминал, цифровой отпечаток, деанонимизация.

1. Введение. Одним из основных параметров, который позволяет уникально идентифицировать абонентов в информационно-телекоммуникационных сетях (ИТКС), является IP-адрес [1, 2], поэтому первоочередная задача методов сохранения анонимности заключается в подмене или сокрытии данного параметра. Для сокрытия IP-адреса используются следующие технологии (рис. 1):

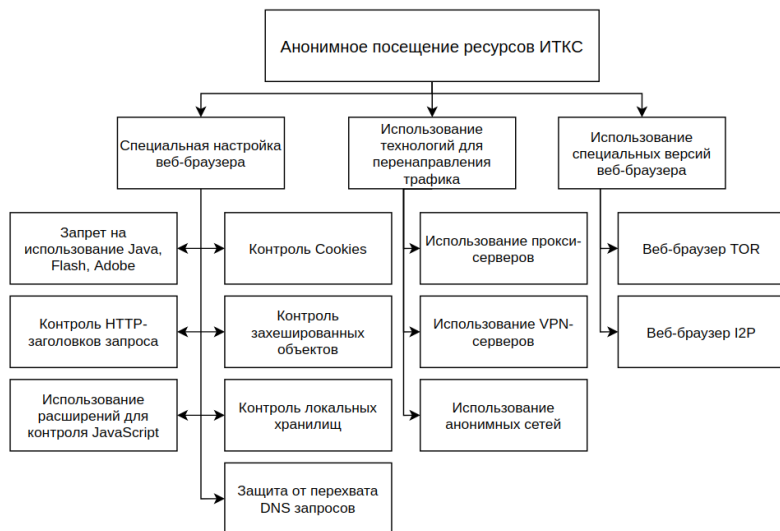


Рис. 1. Технологии анонимизации при передаче данных

– использование прокси-серверов. В этом случае анонимизация доступа к различным ресурсам ИТКС достигается скрытием сведений об источнике запроса или пользователе, при этом целевой получатель видит только IP-адрес и другую информацию прокси-сервера. Возможно использование искажающего прокси-сервера, который передает целевому получателю ложную информацию об истинном адресе источника передаваемых данных;

– виртуальные частные сети – технология, посредством которой создают одно или более соединений поверх другой сети. При этом соединение между абонентом и сервером шифруется, благодаря чему создается максимально анонимный доступ в интернет;

– SSH-тоннель – это технология создания анонимных соединений в ИТКС посредством SSH-протокола. Его особенность в том, что информация шифруется на одном конце тоннеля и расшифровывается на другом. Провайдер и веб-ресурсы здесь используются только для передачи трафика;

– анонимные сети на основе «луковичной маршрутизации» – использующие послойную систему шифрования адресов маршрутизаторов, с помощью которой между пользователем и веб-ресурсами устанавливается анонимное соединение с защитой от перехвата сообщений;

– децентрализованные сети, которые работают поверх основного сетевого протокола ИТКС, при этом в них не используются IP-адреса. Доступ к ним обеспечивается открытым программным обеспечением, создающим специальный тоннель, в котором шифруется трафик и скрываются IP-адреса;

– настройки и плагины к браузерам, которые позволяют отключать функции, отвечающие за сбор сведений об их использовании. Плюсы плагинов для анонимности – простота использования и доступность. Минусы – слабая безопасность и относительная стабильность.

Для обеспечения анонимной передачи данных могут быть использованы сочетания методов обеспечения анонимности (рис. 2).



Рис. 2. Вариант построения комбинированной анонимной сети на базе TOR и VPN

Таким образом, современные технологии предоставляют широкий спектр возможностей по скрытию параметров абонентских терминалов (АТ) при действиях в ИТКС [10, 11], а существующие методики не позволяют идентифицировать АТ анонимных сетей без получения априорных данных о принципах функционирования системы передачи данных.

2. Методы идентификации абонентских терминалов в современных ИТКС. В условиях отсутствия доступа к коммутационно-адресной информации идентифицировать конкретный абонентский терминал ИТКС (а иногда и самого пользователя) можно посредством сбора и классификации доступной информации о параметрах программно-аппаратной реализации абонентских терминалов [18, 19, 21]. Параметры АТ могут быть получены благодаря специфике взаимодействия клиентов и серверов в ИТКС. Так для взаимодействия АТ пользователя с удаленными ресурсами, размещенными на сервере, необходимо специальное программное обеспечение. Основным программным средством для доступа к ресурсам сети интернет является веб-браузер. Схема взаимодействия АТ с удаленным сервером показана на рисунке 3.

Полученные данные формируют цифровой отпечаток – уникальный идентификатор программно-аппаратной конфигурации АТ ИТКС.

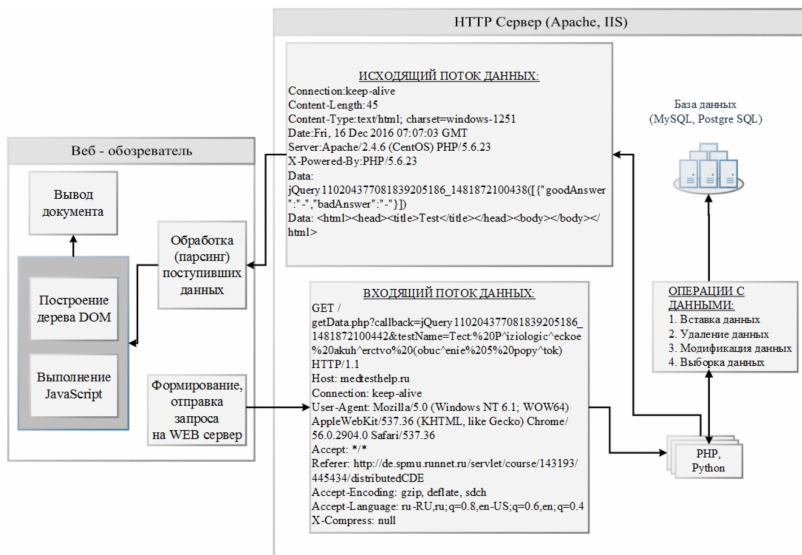


Рис. 3. Схема взаимодействия веб-браузера с удаленными ресурсами на сервере

Существующие методы позволяют получать цифровой отпечаток АТ с помощью сбора максимально доступной информации о настройках операционной системы, версиях и настройках веб-браузеров, установленных драйверах устройств и аппаратных особенностях АТ [3–7]. Классификация методов формирования и хранения цифрового отпечатка представлена на рисунке 4.

Наиболее известный и распространенный способ сохранения небольшого фрагмента данных на машине пользователя – это HTTP cookies, данный способ позволяет гарантированно распознавать и вести статистику обращений АТ к конкретному серверу. Однако в настоящее время существует хорошо отработанная методика защиты от данного способа отслеживания АТ, а именно автоматическая очистка файла, содержащего cookies, использование веб-браузера в режиме «инкогнито», использование специализированных расширений веб-браузера, которые позволяют блокировать и редактировать cookies. Для формирования цифрового отпечатка могут быть использованы различные технологии отображения данных. Так, при использовании флештехнологий (Flash Local Shared Objects) можно сформировать и хранить на АТ уникальный идентификатор веб-браузера.

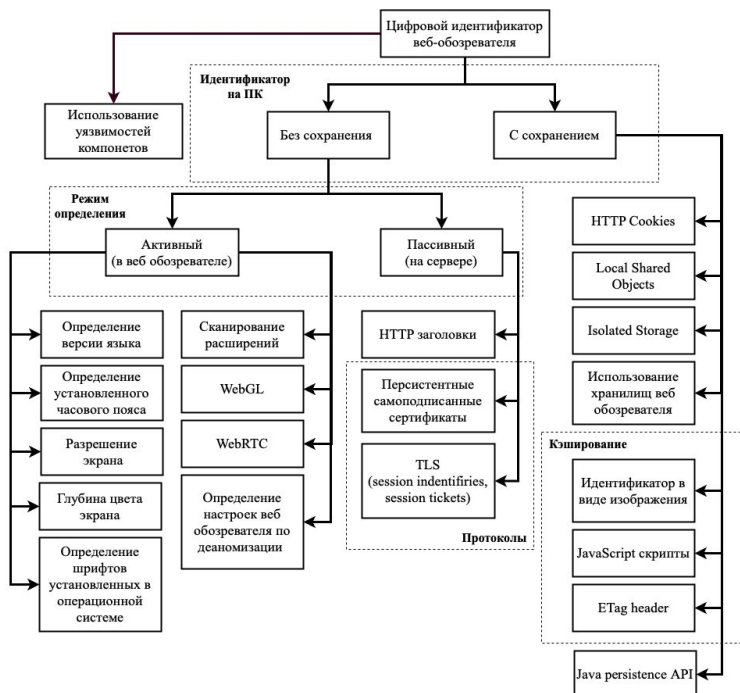


Рис. 4. Классификация способов получения цифрового отпечатка абонентского терминала

Один из самых эффективных пассивных способов формирования и хранения цифрового отпечатка основан на использовании хранилища параметров обозревателя, которые не удаляются при полной очистке и доступны из любых вкладок различных веб-браузеров [17].

Все современные веб-браузеры поддерживают кэширование – временное сохранение файлов на машине пользователя, применяемое для уменьшения нагрузки на сервер. Существуют следующие способы сохранения идентификатора в хранилищах веб-браузера:

- файл JavaScript (идентификатор помещается в переменную);
- изображение (идентификатор помещается внутрь изображения);
- помещение идентификатора в заголовки eTag запрашиваемых у сервера объектов (рис. 5).

Методы снятия цифрового отпечатка без сохранения результатов на АТ можно условно разделить на два режима: активный (в веб-браузере), пассивный (на сервере). Активный режим подразумевает работу в веб-

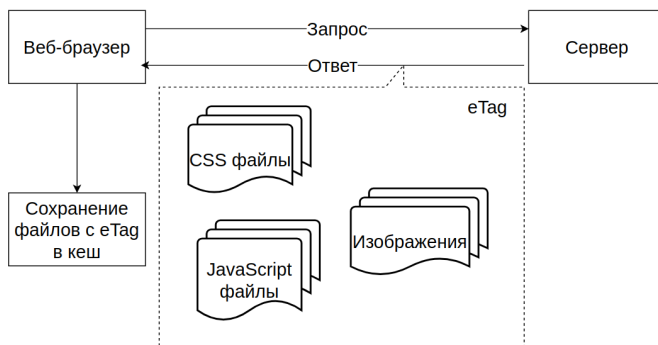


Рис. 5. Схема привязки eTag

браузере исполняемого кода, с помощью которого происходит определение таких характеристик АТ, как:

- версия языка веб-браузера;
- установленный часовой пояс в операционной системе пользователя;
- установленное разрешение экрана персонального компьютера (ПК) пользователя;
- установленная глубина цвета экрана ПК пользователя;
- установленные шрифты в операционной системе пользователя;
- установленные расширения в веб-браузере, в том числе и их версии.

К отдельным способам получения цифрового отпечатка в активном режиме (без сохранения идентификатора на машине пользователя) стоит отнести:

- построение идентификатора посредством анализа особенностей вывода изображения библиотекой WebGL;
- использование уникального идентификатора, создаваемого в WebRTC.

Пассивные режимы определения основаны на поиске или составлении уникального идентификатора на серверной стороне по приходящим данным с ПК пользователя. К ним можно отнести:

- анализ заголовков HTTP-запросов со стороны ПК пользователя к серверу. В результате такого анализа можно определить тип и версию веб-браузера, тип и версию интерпретатора веб-браузера, тип операционной системы пользователя, установленной на ПК, состояние доступа к cookies, состояние кешируемых объектов (на сколько часто происходит удаление);

– персистентные самоподписанные сертификаты, формируемые по запросу серверной стороны, являются уникальным объектом, данные о котором передаются серверу при каждом обращении к нему;

– WebRTC – технология, позволяющая устанавливать соединение между двумя веб-браузерами (соединение типа точка-точка) без управляющих серверов посредников. WebRTC входит в рекомендации консорциума W3C и поддерживается всеми основными браузерами, включая Chrome, Firefox и Opera. Угроза анонимизации с использованием технологии WebRTC заключается в раскрытии реального IP-адреса пользователя;

– идентификация веб-браузера посредством идентификатора кеша session identifiers, session tickets.

Проведенный анализ позволил определить, что наиболее перспективным методом идентификации является активный метод – вследствие устранения возможности подмены личности при посещении ресурсов сети интернет. Однако стоит отметить, что современное антивирусное программное обеспечение проводит мониторинг потенциально уязвимого содержимого, загружаемого в веб-браузер при посещении удаленных ресурсов, кроме того, многие пользователи производят отключение и блокировку уязвимого содержимого, которое позволяет идентифицировать АТ пользователей ИТКС.

Полноценный доступ к современным удаленным ресурсам сети интернет не возможен без использования некоторых технологий. Блокирование технологий веб-браузера приведет к значительным изменениям запрашиваемых данных и, как следствие, к полной невозможности восприятия пользователем полученной информации. Это утверждение относится и к анонимным сетям передачи данных. Такой технологией является использование кода на языке программирования JavaScript как языка сценариев для придания интерактивности веб-страницам.

По состоянию на конец 2018 года примерно 94,5% всех сайтов сети Интернет используют JavaScript для выполнения сценариев на стороне клиента, которые незаметно для пользователя интегрируются в HTML на стороне сервера с помощью кроссплатформенного механизма выполнения.

Таким образом, учитывая механизмы блокировки методов получения цифрового отпечатка, при применении пользователями ИТКС для передачи информации анонимных сетей, использование JavaScript является практически единственным средством сбора информации в интересах дальнейшей идентификации АТ пользователя с применением пассивного метода.

3. Модель системы формирования изображений абонентского терминала. Развитие современных информационно-

коммуникационных сетей влечет за собой создание и модернизацию существующих стандартов и протоколов передачи и представления данных. Так, несколько лет назад была анонсирована и оперативно осуществлена поддержка стандарта HTML5. В настоящее время разработчики активно используют возможности этого стандарта для улучшения уровня поддержки мультимедиа-технологий.

Одним из новых инструментов для расширенных приложений HTML5 является поверхность для рисования Canvas, на которой при помощи кода на JavaScript выполняется вывод графического изображения. При использовании Canvas с сервера загружается не картинка, а набор точек (или алгоритм прорисовки), по которым браузер выполняет вывод изображения.

Таким образом, Canvas – средство программирования, которое позволяет рисовать как простейшие графические примитивы – линии, фигуры, текст, так и создавать различные эффекты мультимедиа.

Рассмотрим модель формирования изображений в браузерах абонентских терминалов современных ИТКС (рис. 6).

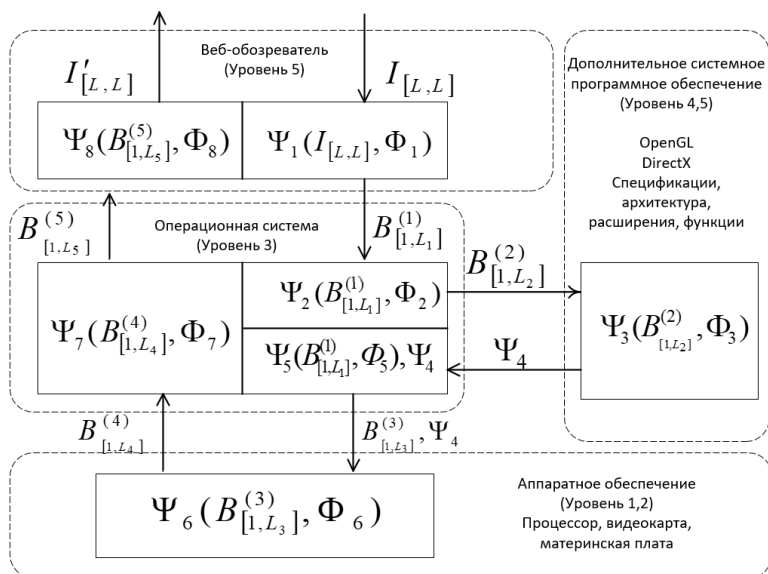


Рис. 6. Модель программно-аппаратной системы обработки графических изображений абонентского терминала ИТКС

При передаче запроса на сервер в браузер поступают данные совместно с кодом на вывод изображения $I_{[L,L]}$ в Canvas. В браузере выполняются преобразования полученных данных для передачи на следующий уровень обработки, при этом учитываются параметры Φ_1 , которые зависят от конкретной реализации веб-браузера (движок браузера).

В любой компьютерной системе взаимодействия прикладных программ с аппаратным обеспечением реализуется при помощи операционной системы.

Браузер с целью получения вычислительных ресурсов для обработки полученного изображения $I_{[L,L]}$, которое представлено исходным кодом шейдера, создает запрос к операционной системе. Запрос содержит обработанные браузером данные и последовательность команд для процессора $B_{[1,L_1]}^{(1)}$. Операционная система преобразует поступившие данные и команды и формирует процедуры взаимодействия с дополнительным системным программным обеспечением $\Psi_2(B_{[1,L_1]}^{(1)}, \Phi_2)$, при этом учитываются функционал и параметры операционной системы Φ_2 . Сформированная последовательность, включающая набор необходимых действий над изображением, и преобразованные данные изображения $B_{[1,L_2]}^{(2)}$ поступают на следующий уровень обработки.

На уровне 4,5 формируются инструкции для аппаратного оборудования $\Psi_3(B_{[1,L_2]}^{(2)}, \Phi_3)$, а именно определяется алгоритм для получения из исходного кода шейдера последовательности данных для вывода графического изображения. При этом на итоговый алгоритм Ψ_4 влияют особенности реализации спецификаций конкретных производителей аппаратной части абонентских терминалов Φ_3 .

Операционная система на основании Ψ_4 формирует последовательность команд Φ_5 для аппаратного оборудования и определяет порядок взаимодействия дополнительного системного программного обеспечения с видеокарты $\Psi_5(B_{[1,L_1]}^{(1)}, \Phi_5)$, Ψ_4 .

Множество возможных реализаций аппаратной части абонентского терминала и отсутствие строгой стандартизации технологического процесса их производства, так как использование комплектующих деталей порождает множество параметров Φ_6 , которые влияют на параметры формируемого графического изображения. При этом производитель может применять фирменные алгоритмы обработки Ψ_6 поступающих данных $B_{[1,L_3]}^{(3)}$ с учетом правила обработки Ψ_4 .

После вычислительных операций $\Psi_6(B_{[1,L_3]}^{(3)}, \Phi_6)$ обработанные данные $B_{[1,L_4]}^{(4)}$ передаются операционной системе, которая, выполнив

соответствующие преобразования $\Psi_7(B_{[1,L_4]}^{(4)}, \Phi_7)$ передает их выполнившему запросу приложению, то есть веб-браузеру. Браузер, в свою очередь, выводит в Canvas итоговое графическое изображение.

Исходя из особенностей формирования цифровых изображений в современных компьютерных системах, множество параметров преобразований накладывают свой отпечаток на выходной графический примитив, формируя тем самым цифровой отпечаток абонентского терминала.

4. Математическая модель цифрового изображения. В современных графических системах формирование цифрового изображения зависит от используемой цветовой модели и может быть представлено как n -мерный массив:

$$I_{[J,L^{(A)},L^{(B)}]}, \quad (1)$$

где J – количество слоев изображения в соответствии с цветовой моделью используемой в графической подсистеме АТ.

Величины $L^{(A)}, L^{(B)}$ определяются размерностью пиксельной сетки выводимого изображения (рис. 7).

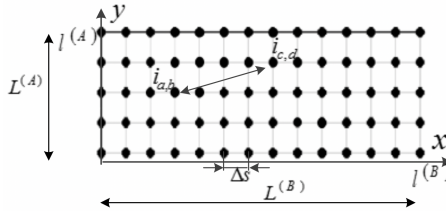


Рис. 7. Представление цифрового изображения в виде массива пикселей

В частном случае для одного слоя изображения функция $I_{[L^{(A)},L^{(B)}]}$ определяется двумерным массивом. Для простоты расчетов в работе все изображения определяются массивами $I_{[L,L]}$ ($L^{(A)} = L^{(B)} = L$):

$$I_{[L,L]} = \begin{bmatrix} i_{1,1} & i_{1,2} & \dots & i_{1,L} \\ i_{2,1} & i_{2,2} & \dots & i_{2,L} \\ \dots & \dots & \dots & \dots \\ i_{L,1} & i_{L,2} & \dots & i_{L,L} \end{bmatrix}, \quad (2)$$

где $i_{l_1,l_2}, l_1 \in (1, \dots, L), l_2 \in (1, \dots, L)$ – значение соответствующего пикселя изображения.

Для отображения цветных изображений пиксели в современных компьютерных системах разделяют на триады (для цветовой модели RGB – красного, зеленого и синего цветов), которые в совокупности формиру-

ют один результирующий цвет пикселя. Для отрисовки изображения в графической подсистеме цвет каждого элемента задается путем выборки некоторого значения (индекса) из специальной таблицы — палитры цветов.

Каждая триада пикселя имеет в качестве цвета некий условный индекс, который расшифровывается по палитре цветов в реальные компоненты цвета, что позволяет для каждого такого элемента построить «истинный» цвет в заданной цветовой модели. Модель цифрового изображения графической системы современного компьютера состоит из определенного количества слоев и зависит от многомерной функции пространственного распределения значений яркости и прозрачности трех цветов:

$$I_{[J,L,L]} = (i_{j,l(x),l(y)})_{j=1,x=1,y=1}^{4,A,B}, \quad (3)$$

где $j \in 1, \dots, J$ – множество слоев изображения, для $RGB\alpha$ $J = 4$, A, B – размер изображения в соответствии с выражением (1) (R – красный, G – зеленый, B – синий, α – альфа-канал), $i_{1,l(x),l(y)} \in \{0, \dots, 255\}$, $i_{2,l(x),l(y)} \in \{0, \dots, 255\}$, $i_{3,l(x),l(y)} \in \{0, \dots, 255\}$, $i_{4,l(x),l(y)} \in [0; 1]$ – значения альфа-канала (0 – полностью прозрачный, 1 – не прозрачный).

Таким образом, цифровое изображение в современных компьютерных системах представляет собой многомерный массив значений палитры цветов. Простейшим элементом изображения является пиксель. Объединение пикселей различных цветов в некоторой области дает результирующий цвет в соответствии с применяемыми алгоритмами обработки и используемой цветовой моделью.

5. Алиасинг цифровых изображений. Использование подхода, который предполагает представление цифровых изображений как двухмерных массивов пикселей, может привести к проблеме, называемой алиасингом. Алиасинг в компьютерной графике – это эффект «ступенчатости» изображения (рис. 8).

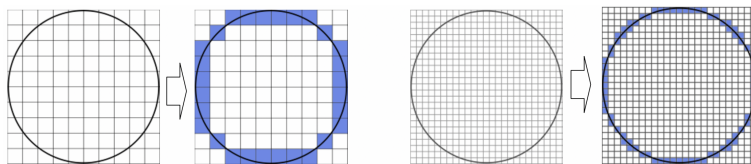


Рис. 8. Влияние эффекта алиасинга на точность представления аналоговой информации в цифровой форме

Для устранения эффекта алиасинга применяются алгоритмы сглаживания (антиалиасинга), идея которых заключается в преобразовании

цифрового изображения в удобном для восприятия зрительной системой человека виде. В общем виде результат применения метода сглаживания представлен на рисунке 9.

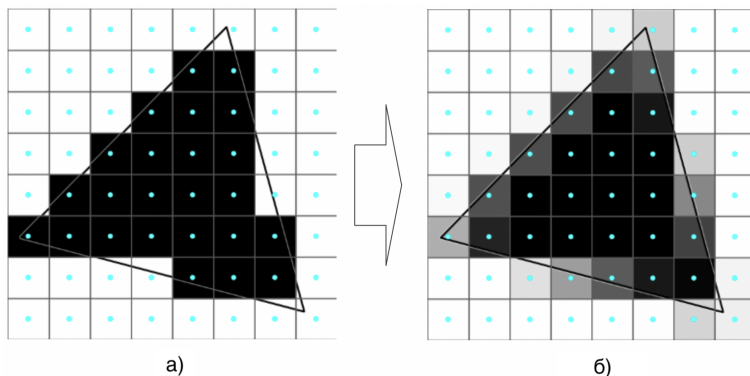


Рис. 9. Устранение эффекта алиасинга: а) изображение с алиасингом; б) изображение при применении алгоритма антиалиасинга

Из рисунка (9) видно, что при ступенчатом выводе границ изображения решение по цвету пикселя применяется исходя из положения центра пикселя относительно геометрии вывода изображения и принимает два значения, в то время как цвет пикселя с учетом антиалиасинга определяется в зависимости от функции расстояния до границы изображения и варьируется от 0 до 255.

Классификация видов антиалиасинга и реализующие их алгоритмы широко представлены в технической литературе [8,9,20], а также частично представлены программно-аппаратной документацией производителей комплектующих АТ.

6. Математическая модель системы сглаживания изображений в современных АТ ИТКС. Основным принципом сглаживания изображений в современных компьютерных системах является вычисление цветов пикселей в граничных участках графического отображения фигур и объектов. При этом алгоритмы этих вычислений основываются на особенностях зрительного восприятия графических отображений человеком.

С целью формирования итогового цвета для каждого пикселя необходимо сформировать решетку размещения семплов $G_{[H,H]}$ со сторонами равными H (рис. 10). После чего определить число семплов A и их положение на сформированной решетке. Элементы массива $G_{[H,H]}$ могут

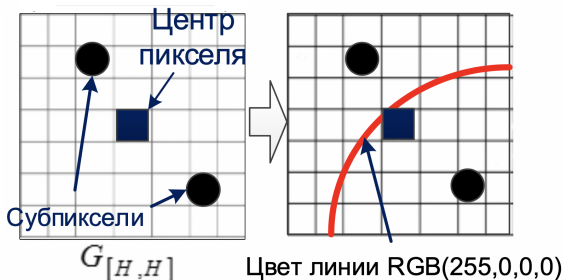


Рис. 10. Расположение семплов в пикселе

быть получены следующим образом:

$$g_{h_1, h_2} = \begin{cases} 1 & \text{для } h_1 \in X^{(G)}, h_2 \in Y^{(G)}; \\ 0 & \text{для } h_1 \notin X^{(G)}, h_2 \notin Y^{(G)}, \end{cases} \quad (4)$$

где $h_1 = 1(1)H, h_2 = 1(1)H$.

Размещение семплов зависит от координат:

$$X^{(G)} = \{x_1^{(G)}, \dots, x_A^{(G)}\}, Y^{(G)} = \{y_1^{(G)}, \dots, y_A^{(G)}\}, \quad (5)$$

где A – число семплов в пикселе.

Число семплов и их расположение зависит от аппаратной части графической подсистемы АТ ИТКС. Каждый производитель оборудования может устанавливать собственные параметры размера решетки и числа семплов, используемых для сглаживания изображений. Множество всех возможных значений числа семплов в пикселе можно представить как $A = \{2^0, 2^1, \dots, 2^n\}$. В современных графических устройствах, как правило, используется от 2 до 16 семплов на пиксель.

Существует несколько вариантов формирования итогового цвета пикселя. Один из таких вариантов основывается на числе семплов, попавших в область объекта $\Omega^{(\text{ОБЪЕКТ})}$, при этом значение цвета может быть представлено следующим образом (рис. 11):

$$C^{(\text{результат})} = \frac{N^{(\text{фигуры})}}{N^{(\text{пиксель})}} \times C^{(\text{фигуры})}, \quad (6)$$

где $C^{(результат)}$ – результирующий цвет пикселя; $C^{(фигуры)}$ – цвет фигуры; $N^{(фигуры)}$ – число семплов, попавших в область фигуры; $N^{(пиксель)}$ – общее число семплов в пикселе.

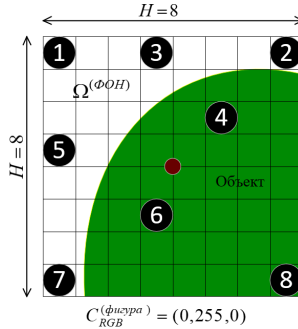


Рис. 11. Пример размещения семплов в пикселе при формировании объекта изображения

В приведенном на рисунке 11 примере результирующий цвет пикселя может быть рассчитан как:

$$C_{RGB}^{(пиксель)} = \frac{3}{8} \times (0, 255, 0) = (0, 95, 0). \quad (7)$$

При этом видно, что объект занимает большую часть пикселя, однако число пикселей, попавших в область объекта, меньше числа семплов фона, поэтому результирующий цвет будет не в полной мере влиять на сглаживание краев изображения. Для устранения этого эффекта используют такой параметр как расстояние от центра пикселя до семпла (рис. 12).

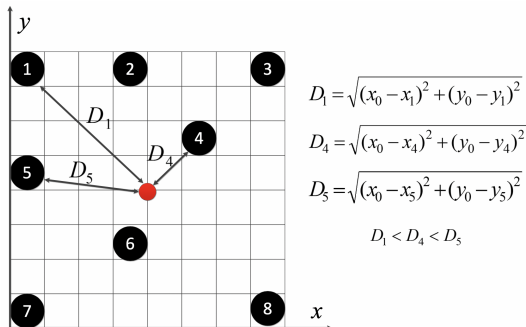


Рис. 12. Неоднородность расстояния между центром пикселя и семплами

Цвет пикселя в этом случае зависит от значений расстояний до семплов входящих в область объекта и может быть вычислен как:

$$C^{(\text{результат})} = \frac{\sum_{n=1}^{N(\text{фигуры})} k_n D_n}{\sum_{m=1}^{N(\text{пиксель})} k_m D_m} \times C^{(\text{фигуры})}, \quad (8)$$

где D_n – расстояние от семплов до центра пикселя; D_m – расстояние от всех семплов до центра пикселя; k_n – коэффициент влияния семпла на цвет относительно положения.

Производители вправе самостоятельно определять влияние расстояний на определение итогового цвета. Поэтому возможны следующие варианты задания коэффициентов k_n (рис. 13).

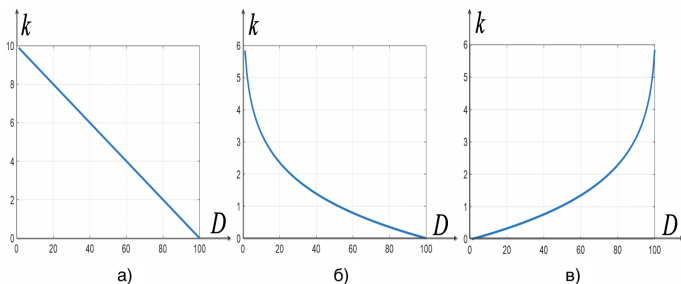


Рис. 13. Зависимость коэффициента значимости семпла от расстояния: а) линейная убывающая зависимость; б) нелинейная убывающая зависимость; в) нелинейная возрастающая зависимость

Коэффициенты значимости семпла могут быть представлены как массив значений $V_{[H,H]} = ||v_{h_1,h_2}||$ ($h_1 = 1(1)H, h_2 = 1(1)H$), где v_{h_1,h_2} – значение коэффициента значимости; H – размер массива коэффициентов.

Формирующая последовательность для массива коэффициентов значимости может быть представлена как $W = \langle w_1, w_2, \dots, w_H \rangle = \langle \beta_1 \times \omega, \beta_2 \times \omega, \dots, \beta_H \times \omega \rangle$, где $B = \langle \beta_1, \beta_2, \dots, \beta_H \rangle$ – коэффициенты, задающие тип последовательности (линейный, нелинейный, убывающий, возрастающий), $\omega = 1(1)H$.

Таким образом, итоговый массив коэффициентов значимости может быть получен как:

$$V_{[H,H]} = ||v_{h_1,h_2}||, \text{ где } v_{h_1,h_2} = f(w) = w_{h_1} \times w_{h_2}. \quad (9)$$

Пример массива коэффициентов значимости представлен на рисунке 14.

$$W = (1, 2, 3, 4, 5, 6, 6, 5, 4, 3, 2, 1)$$

1	2	3	4	5	6	6	5	4	3	2	1
2	4	6	8	10	12	12	10	8	6	4	2
3	6	9	12	15	18	18	15	12	9	6	3
4	8	12	16	20	24	24	20	16	12	8	4
5	10	15	20	25	30	30	25	20	15	10	5
6	12	18	24	30	36	36	30	24	18	12	6
6	12	18	24	30	36	36	30	24	18	12	6
5	10	15	20	25	30	30	25	20	15	10	5
4	8	12	16	20	24	24	20	16	12	8	4
3	6	9	12	15	18	18	15	12	9	6	3
2	4	6	8	10	12	12	10	8	6	4	2
1	2	3	4	5	6	6	5	4	3	2	1

$$W = (1, 2, 4, 8, 14, 25, 25, 14, 8, 4, 2, 1)$$

1	2	4	8	14	25	25	14	8	4	2	1
2	4	8	16	28	50	50	28	16	8	4	2
4	8	16	32	56	100	100	56	32	16	8	4
8	16	32	64	112	200	200	112	64	32	16	8
14	28	56	112	196	350	350	196	112	56	28	14
25	50	100	200	350	625	625	350	200	100	50	25
25	50	100	200	350	625	625	350	200	100	50	25
14	28	56	112	196	350	350	196	112	56	28	14
8	16	32	64	112	200	200	112	64	32	16	8
4	8	16	32	56	100	100	56	32	16	8	4
2	4	8	16	28	50	50	28	16	8	4	2
1	2	4	8	14	25	25	14	8	4	2	1

$$W = (25, 14, 8, 4, 2, 1, 1, 2, 4, 8, 14, 25)$$

625	350	200	100	50	25	25	50	100	200	350	625
350	196	112	56	28	14	14	28	56	112	196	350
200	112	64	32	16	8	8	16	32	64	112	200
100	56	32	16	8	4	4	8	16	32	56	100
50	28	16	8	4	2	2	4	8	16	28	50
25	14	8	4	2	1	1	2	4	8	14	25
25	14	8	4	2	1	1	2	4	8	14	25
50	28	16	8	4	2	2	4	8	16	28	50
100	56	32	16	8	4	4	8	16	32	56	100
200	112	64	32	16	8	8	16	32	64	112	200
350	196	112	56	28	14	14	28	56	112	196	350
625	350	200	100	50	25	25	50	100	200	350	625

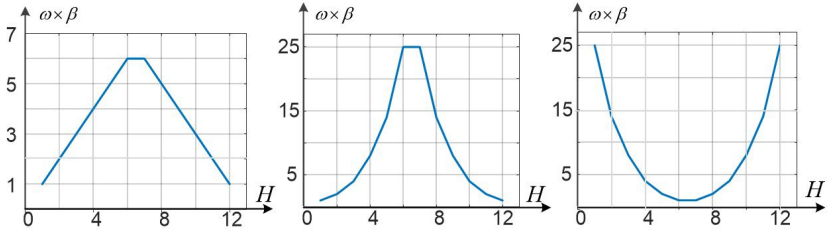


Рис. 14. Массив коэффициентов значимости

На основании массивов размещения семплов $G_{[H,H]}$ и массива коэффициентов значимости $V_{[H,H]}$ формируется массив семплов $R_{[H,H]}$, в котором каждый семпл обладает определенным весом $R_{[H,H]} = G_{[H,H]} \times V_{[H,H]}$.

Пример формирования весового массива семплов представлен на рисунке 15.

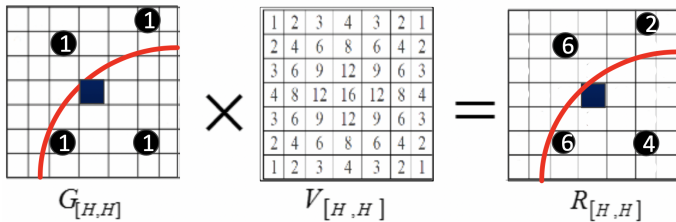


Рис. 15. Формирование весового массива семплов

Степень влияния семплов массива $R_{[H,H]}$ может быть представлена массивом $P_{[H,H]}$, элементы которого отражают долю влияния каждого семпла на итоговый цвет пикселя и рассчитывается следующим образом:

$$p_{h_1, h_2} = \frac{r_{h_1, h_2}}{\sum_{n_1=1}^H \sum_{n_2=1}^H r_{n_1, n_2}}, r_{h_1, h_2} \in R_{[H, H]}. \quad (10)$$

При этом массив семплов, попавших в область объекта изображения, может быть получен как:

$$p_{h_1, h_2}^{(C)} = \begin{cases} p_{h_1, h_2}, & \text{при } g_{h_1, h_2} \in \Theta \\ 0, & \text{при } g_{h_1, h_2} \notin \Theta \end{cases}, h_1 = 1(1)H, h_2 = 1(1)H. \quad (11)$$

где Θ – область объекта изображения.

Итоговый цвет пикселя C' в этом случае получается в результате суммирования по строкам и столбцам всех значений массива семплов, попавших в область объекта изображения Θ :

$$C' = C \times \sum_{h_1=1}^H \sum_{h_2=1}^H p_{h_1, h_2}^{(C)}, \quad (12)$$

где C – исходный цвет объекта изображения.

На рисунке 16 представлен процесс формирования итогового цвета пикселя.

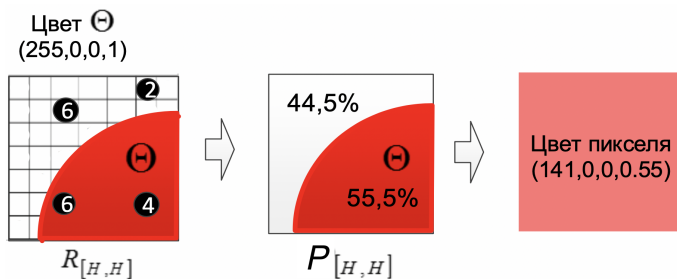


Рис. 16. Формирование весового массива семплов

В зависимости от производителя графической системы и комплекта программных средств вычисление результирующих значений цвета пикселей могут дополняться более сложными алгоритмами. Так современные АТ ИТКС используют более 50 различных алгоритмов сглаживания.

Таким образом, цифровой отпечаток устройства зависит от размещения семплов в пикселе, от используемых алгоритмов вычисления

степени влияния пикселей, а также от применяемых процедур сглаживания изображений в графической подсистеме.

7. Информативный признак неоднородности программно-аппаратных реализаций АТ ИТКС. При формировании изображений результаты сглаживания в различных реализациях графических систем АТ ИТКС имеют разную степень отличия. Чтобы выявить отличия необходимо последовательно сравнивать значения каждого пикселя многомерного массива изображений $I_{[J,L,L]}$, формируемого терминалом с эталоном изображения полученным без эффекта сглаживания.

Для выявления отличий в многомерных массивах $I_{[J,L,L]}^{(AT)}$ каждый j -ый массив многомерного массива представляется в виде последовательности значений:

$$Z^{(j)} = \langle Z_1^{(j)}, Z_2^{(j)}, \dots, Z_L^{(j)} \rangle, \tag{13}$$

где $Z_n^{(j)} = \langle i_{1,n}, i_{2,n}, \dots, i_{L,n} \rangle$ – строка массива $I_{[j,L,L]}$ из многомерного массива $I_{[J,L,L]}$, $n = 1(1)L$.

Полученные последовательности для цветовой модели RGBA являются трансформацией представления цветов из двумерного пространства в одномерное. Изображение формируется из последовательности значений цветов пикселей – так называемая кадровая развертка (рис. 17). Поэтому перевод из массива в строку не противоречит самому принципу формирования цифровых изображений.

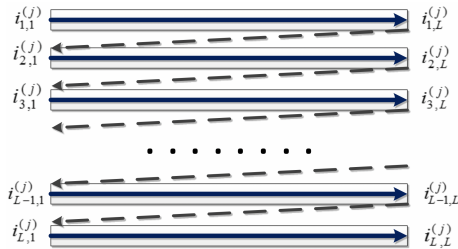


Рис. 17. Формирование цифрового изображения из последовательности значений цветов пикселей

Далее производится конкатенация значений цветов пикселей, полученных из массивов каждого цвета, и альфа-канала:

$$Z^{(\Sigma)} = \langle Z^{(1)} \oplus Z^{(2)} \oplus Z^{(3)} \oplus Z^{(4)} \rangle, \tag{14}$$

где \oplus – операция конкатенации последовательностей.

Для выявления особенностей формирования цифрового изображения конкретной графической системой необходим эталон для сравнения полученных последовательностей от разных систем. За такой эталон может быть принята последовательность, полученная из цифрового изображения, которое сформировано без учета сглаживания.

Пусть E последовательность, полученная из отображения графической информации без учета антиалиасинга и не учитывающая палитру цветов:

$$E = \langle e_1, e_2, \dots, e_{L \cdot L \cdot 4} \rangle, \quad (15)$$

тогда последовательность $U = Z(\Sigma) - E$ будет включать набор значений, которые свидетельствуют об особенностях реализации графической подсистемы АТ ИТКС, так как учитывает результаты сглаживания пересекающихся объектов изображения, которые зависят от определенного набора программно-аппаратных средств вывода графических примитивов.

Для информативного графического представления данных будем использовать контрольную карту кумулятивных сумм (ККС), которая предназначена для проверки на отклонение от некоторого эталонного значения. На такой карте информативны не абсолютные значения сумм, а угол наклона графика, определяемый по последовательным точкам, и его форма.

Метод кумулятивных сумм может быть применен при анализе данных для обнаружения любых различий между множеством входных последовательностей $k_\varepsilon = \sum_{n_1}^{\varepsilon} (e_n - z_n) = \sum_{n_1}^{\varepsilon} u_n$ ($\varepsilon = 1(1)L(\Sigma)$). Элементы индикаторной последовательности, полученные с помощью применения метода кумулятивных сумм:

$$K = \langle k_1, k_1, \dots, k_{L(\Sigma)} \rangle, \quad (16)$$

где $L(\Sigma) = L \cdot L \cdot 4$ – длина индикаторной последовательности.

В результате проведенных экспериментальных вычислений с использованием разработанной модели были выявлены особенности построения графиков индикаторных последовательностей при различных параметрах графических подсистем АТ ИТКС (рис. 18).

Для рисунков 18-23 эксперименты проводились на одном изображении, представляющем собой наложенные друг на друга геометрические фигуры с пересечением границ.

Из рисунка 18 следует, что индикаторные последовательности графических подсистем АТ ИТКС с различными параметрами аппаратной

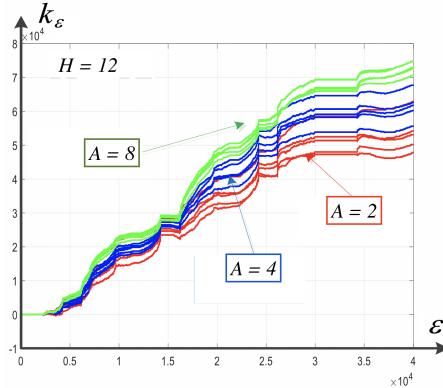


Рис. 18. Значения индикаторной последовательности при переменном числе семплов A в пикселе формируемого изображения

реализации имеют отличные углы наклона относительно координатных осей.

Расчеты производились при размере решетки семплов $H = 14$, размере пиксельной решетки $L = 200$, количестве слоев изображения $\Phi = 9$, палитре цветов $C = \{0, \dots, 255\}$.

Таким образом, возможно распознавание АТ ИТКС по особенностям определения итогового цвета пикселя, которое зависит от числа и размещения семплов в пикселе, так как в индикаторных последовательностях прослеживаются особенности графических подсистем.

На рисунке 19 показаны значения индикаторной последовательности при разных значениях размера решетки семплов, количестве семплов $A = 4$, расчет влияния семплов на итоговый цвет производился по одному алгоритму для всех вариантов формирования цифрового изображения.

Исходя из представленных на рисунок 19 зависимостей можно сделать вывод, что размер решетки семплов оказывает существенное влияние на формирование значений индикаторной последовательности и является параметром, который позволит выявить особенность формирования графического примитива конкретным АТ ИТКС.

Для исследования влияния на индикаторную последовательность минимального сдвига семпла в структуре пикселя были проведены вычислительные эксперименты. Вычисления проводились для разных размеров решетки семплов, число семплов $A = 4$, $L = 200$, $C = \{1, \dots, 255\}$.

Результаты экспериментов приведены на рисунке 20.

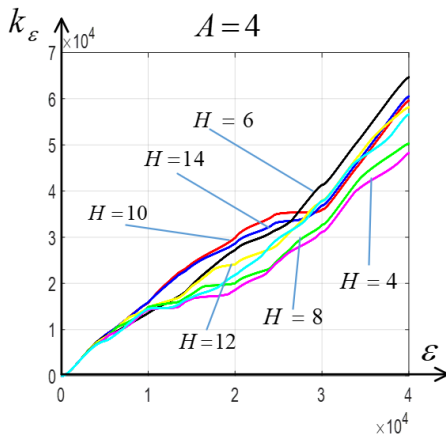


Рис. 19. Значения индикаторной последовательности при разном размере семпла

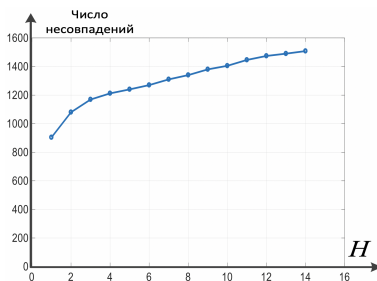


Рис. 20. Зависимость числа несовпадений двух индикаторных последовательностей при сдвиге одного семпла

В качестве меры отличия между индикаторными последовательностями принимается число пикселей, в которых отличался цвет изображений. Таким образом, минимальное изменение положения одного семпла оказывает значительное влияние на форму и угол наклона индикаторной последовательности, это влияние усиливается при увеличении размера решетки семплов.

При исследовании зависимости значений индикаторной последовательности от размеров пиксельной решетки выявлено, что размер пиксельной решетки не оказывает влияния на вид индикаторной последовательности. Таким образом, можно использовать изображения с более низким разрешением, что повысит оперативность обработки массивов цветовой модели и при этом не повлияет на качество распознавания.

При исследовании влияния количества слоев изображения на вид индикаторной последовательности установлена зависимость числа отличий цветов пикселей от числа слоев (рис. 21).

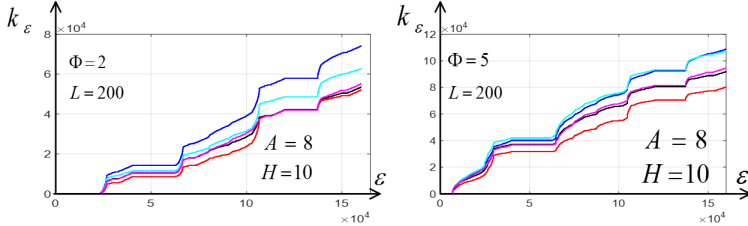


Рис. 21. Зависимость значений индикаторной последовательности от числа слоев изображения: а) два слоя; б) пять слоев

В качестве показателя отличий использовался средний коэффициент невязки всех индикаторных последовательностей, который можно представить следующим выражением:

$$\alpha = \frac{1}{2N^{(K)}} \sum_{i=1}^{N^{(K)}} \sum_{j=1}^{N^{(K)}} \left(\frac{1}{L^{(\Sigma)}} \sum_{n=1}^{L^{(\Sigma)}} (K_i - K_j) \right), \quad (17)$$

где i, j – номера участвующих в вычислительном эксперименте индикаторных последовательностей; K_i, K_j – индикаторные последовательности; $N^{(K)}$ – число индикаторных последовательностей.

Зависимость среднего коэффициента невязки от числа слоев изображения представлена на рисунке 22.

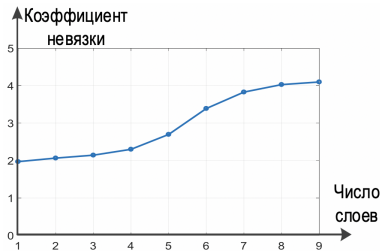


Рис. 22. Зависимость среднего коэффициента невязки от числа слоев изображения

При исследовании влияния палитры цветов на индикаторную последовательность получены следующие результаты (рис. 23).

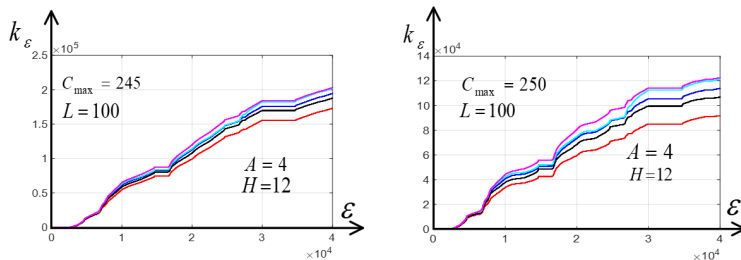


Рис. 23. Зависимость значений индикаторной последовательности от используемой палитры цветов

Таким образом, палитра цветов оказывает влияние на вид индикаторной последовательности. Так как палитра зависит от программно-аппаратной части графической системы, то получаемые значения являются отображениями реализации программно-аппаратной системы АТ ИТКС, что может использоваться в идентификации конкретного АТ.

8. Проведение экспериментальных исследований. Цель экспериментальных исследований состоит в анализе модели формирования изображения, учитывающей программно-аппаратную часть АТ. В рамках проведенных теоретических исследований и на основе модели формирования изображения разработан специализированный программный комплекс (СПК).

Работа СПК оценивалась на контрольном примере. В качестве АТ выбрано две группы ЭВМ:

1. Первая группа базируется на одинаковом программно-аппаратном составе, за исключением используемых видеоадаптеров.
2. Вторая группа базируется на одинаковом программно-аппаратном составе, за исключением используемого веб-браузера.

В качестве отображаемых веб-браузером графических примитивов используется изображение размерности 200×200 пикселей. На основе разработанного программного комплекса произведен расчет информативного признака для каждого АТ. На рисунке 24 показаны различия вывода изображений при сравнении с эталоном, линии представляют собой кумулятивную сумму, которая является значением информативного признака.

При проведении экспериментов в рамках второй группы АТ (одинаковый программно-аппаратный состав, за исключением используемого веб-браузера) были получены различия при выводе изображений (рис. 25), что подтверждает работоспособность метода идентификации АТ.

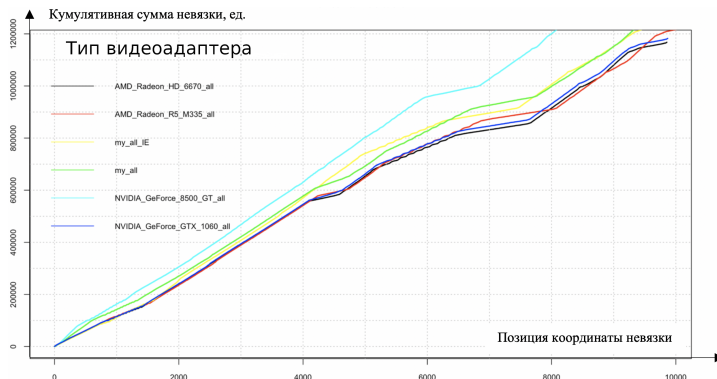


Рис. 24. Различия вывода изображения при сравнении с эталоном для первой группы (аппаратные отличия АТ)

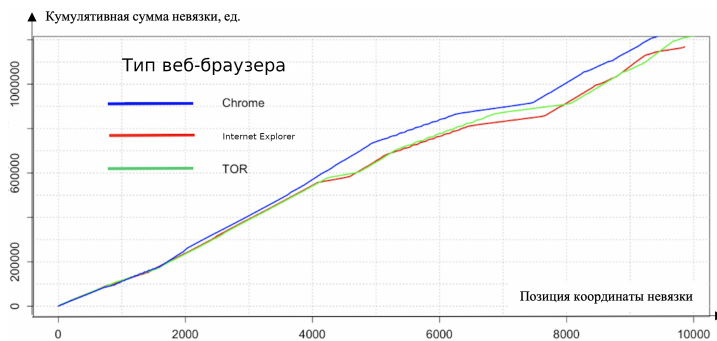


Рис. 25. Различия вывода изображения при сравнении с эталоном для второй группы (программные отличия АТ)

9. Заключение. Разработана аналитическая модель графической подсистемы формирования изображений в АТ ИТКС.

На основе разработанной модели сформирован информативный признак цифровых изображений АТ ИТКС, а именно угол наклона и форма индикаторной последовательности, которая, в свою очередь, служит мерой изменения случайной величины. Чем больше угол наклона линии, тем больше различие двух последовательностей значений цветов пикселей цифрового изображения.

Обоснованы особенности формирования цифровых изображений в графических подсистемах современных компьютерных систем. Данные особенности позволяют осуществлять идентификацию в условиях апри-

орной неопределенности относительно режимов и параметров передачи информации, а также создать условия для устранения неопределенности относительно принадлежности АТ ИТКС [12–16, 22].

Для информативного признака рассмотрены условия его проявления, что, в целом, дает возможность рассмотреть области его применимости.

Литература

1. *Invernizzi L. et al.* Cloak of visibility: detecting when machines browse a different web // Proceedings of the 37th IEEE Symposium on Security and Privacy. 2016. pp. 743–758.
2. *Mowery K., Bogenreif D., Yilek S., Shacham H.* Fingerprinting information in JavaScript implementations // Proceedings of W2SP. 2011. vol. 2. no. 11.
3. *Pham K., Santos A.S.R., Freire J.* Understanding website behavior based on user agent // Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2016. pp. 1053–1056.
4. *Torres C.F., Jonker H., Mauw S.* FP-Block: usable web privacy by controlling browser fingerprinting // European Symposium on Research in Computer Security. 2015. pp. 3–19.
5. *Xu H., et al.* Detecting and characterizing web bot traffic in a large e-commerce marketplace // European Symposium on Research in Computer Security. 2018. vol. 11099. pp. 143–163.
6. *Navpreet K. et al.* Browser Fingerprinting as user tracking technology // 2017 11th International Conference on Intelligent Systems and Control (ISCO). 2017. pp. 103–111.
7. *Queiroz J.S., Feitosa E.L.* A Web Browser Fingerprinting Method Based on the Web Audio API // The Computer Journal. 2019. vol. 62. no. 8. pp. 1106–1120.
8. *Carpenter L.* The A-buffer an Antialiased Hidden Surface Method // IEEE Computer Graphics and applications. 1984. vol. 18. no. 3. pp. 103–108.
9. *Crow F. C.* Comparison of Antialiasing Techniques // IEEE Computer Graphics and applications. 1981. vol. 1. no. 1. pp. 40–48.
10. *Сазонов К.В., Татарка М.В., Шуваев Ф.Л.* Метод структурного статистического распознавания режимов работы информационно-вычислительных сетей // Научно-технический журнал «Информация и космос». 2019. № 1. С. 40–46.
11. *Сазонов К.В., Татарка М.В., Шуваев Ф.Л.* Метод сокращения размерности вероятностных суффиксных деревьев // Информация космос. 2018. № 3. С. 55–61.
12. *Flach P.* Machine learning: the art of science of algorithms that make sense of data // Cambridge University press. 2012. 400 p.
13. *Jordan M.I., Mitchell T.M.* Machine learning: Trends, perspectives, and prospects // Science. 2015. vol. 349(6245). pp. 255–260.
14. *Айма V.A. et al.* Classification algorithms for big data analysis a map reduce approach // The International Archives of the Photogrammetry" in Remote Sensing and Spatial Information Sciences. 2015. vol. 40. no. 3. pp 17.
15. *Grolinger K. et al.* Challenges for MapReduce in Big Data // IEEE 10th World Congress on Services. 2014. pp. 182–189.
16. *Dhillon S., Kaur K.* Comparative Study of Classification Algorithms for Web Usage Mining // International Journal of Advanced Research in Computer Science and Software Engineering. 2014. vol. 4(7). pp. 137–140.
17. *Nikiforakis N. et al.* Cookieless monster: Exploring the ecosystem of web- based device fingerprinting // 2013 IEEE Symposium on Security and Privacy (SP). 2013. pp. 541–555.

18. *Khan T.M., Khan M.A.U., Kong Y.* Fingerprint image enhancement using multi-scale DDFB based diffusion filters and modified Hong filters // *Optik*. 2014. vol. 125. no. 16. pp. 4206–4214.
19. *Khan M.A.U., Khan T.M., Bailey D.G., Kong Y.* A spatial domain scar removal strategy for fingerprint image enhancement // *Pattern Recognition*. 2016. vol. 60. pp. 258–274.
20. *Khan T.M., Khan M.A.U., Kittaneh O., Kong Y.* Stopping criterion for linear anisotropic image diffusion: A fingerprint image enhancement case // *EURASIP Journal on Image and Video Processing*. 2016. vol. 2016. no. 1. pp. 6.
21. *Dunbar C., Qu G.* A practical circuit fingerprinting method utilizing observability don't care conditions // *Proceedings of the 52nd Annual Design Automation Conference*. 2015. pp. 1–6.
22. *Dunbar C., Qu G.* Satisfiability don't care condition based circuit fingerprinting techniques // *The 20th Asia and South Pacific Design Automation Conference (ASPDAC'15)*. 2015. pp. 815–820.

Сазонов Константин Викторович — д-р техн. наук, доцент, начальник кафедры, Военно-космическая академия имени А.Ф. Можайского (ВКА им. А.Ф. Можайского). Область научных интересов: теория вероятности и математическая статистика, распознавание образов, представление и обработка данных и знаний. Число научных публикаций — 40. Staffa78@mail.ru; ул. Ждановская, 13, 197198, Санкт-Петербург, Россия; р.т.: +7(812)347-97-70; факс: +7(812)237-12-49.

Татарка Максим Васильевич — канд. техн. наук, докторант, Военно-космическая академия имени А.Ф. Можайского (ВКА им. А.Ф. Можайского). Область научных интересов: теория вероятности и математическая статистика, распознавание образов, анализ больших объемов данных. Число научных публикаций — 22. maksimtbv@gmail.com; ул. Ждановская, 13, 197198, Санкт-Петербург, Россия; р.т.: +7(812)347-97-70; факс: +7(812)237-12-49.

Цыганов Андрей Сергеевич — канд. техн. наук, старший преподаватель, Военно-космическая академия имени А.Ф. Можайского (ВКА им. А.Ф. Можайского). Область научных интересов: теория вероятности и математическая статистика, распознавание образов, технический анализ сигналов. Число научных публикаций — 18. porudchik@mail.ru; ул. Ждановская, 13, 197198, Санкт-Петербург, Россия; р.т.: +7(812)347-97-70; факс: +7(812)237-12-49.

Бессольцев Виталий Евгеньевич — преподаватель, Военно-космическая академия имени А.Ф. Можайского (ВКА им. А.Ф. Можайского). Область научных интересов: теория вероятности и математическая статистика, распознавание образов, веб-разработка. Число научных публикаций — 8. v.bessoltsev@gmail.com; ул. Ждановская, 13, 197198, Санкт-Петербург, Россия; р.т.: +7(812)347-97-70; факс: +7(812)237-12-49.

K. SAZONOV , M. TATARKA , A. TSYGANOV , V. BESSOLCEV
**IDENTIFICATION OF SUBSCRIBER TERMINALS OF
INFOCOMMUNICATION NETWORKS BASED ON THE MODEL
OF FORMING IMAGES IN MODERN COMPUTER SYSTEMS**

Sazonov K., Tatarka M., Tsyganov K., Bessolcev V. **Identification of Subscriber Terminals of Infocommunication Networks based on the Model of Forming Images in Modern Computer Systems.**

Abstract. One of the important tasks of such theories as theories of pattern recognition and the theory of information security, is the task of identifying terminals of information and telecommunication networks.

The relevance of the topic is due to the need to study methods for identifying computer network terminals and build information security systems based on the knowledge gained

The main parameters that allow uniquely identifying subscriber terminals in the network are address-switching information, as well as a number of parameters characterizing the software and hardware of the computer system. Based on the obtained parameters, digital fingerprints of subscriber terminals are generated.

The using anonymous networks by users of subscriber terminals and blocking of the methods of generating and collecting digital fingerprint parameters, does not allow to achieve the required degree of identification reliability in some cases.

Based on the peculiarities of digital image formation in modern computer systems, many transformation parameters make impact on the output graphic primitive, thereby forming a digital fingerprint of the subscriber terminal, which depends on the placement of samples in a pixel, the algorithms used to calculate the degree of pixels influence, and also the procedures used of smoothing images in the graphics subsystem.

In this paper an original model of image formation by means of a subscriber terminal web browser that allows to increase the degree of reliability of identification under conditions of anonymization of users of information and telecommunication networks is proposed.

Features of the digital images formation in the graphic subsystems of modern computer systems are substantiated. These features allow identification under a priori uncertainty regarding the modes and parameters of information transfer.

Keywords: Identification, Imaging Model, Subscriber Terminal, Digital Fingerprint, Deanonymization.

Sazonov Konstantin — Ph.D., Dr.Sci., Associate Professor, Head of Department, Mozhaisky Military Space Academy. Research interests: probability theory and mathematical statistics, pattern recognition, data and knowledge representation and processing. The number of publications — 40. Staffa78@mail.ru; 13, Zhdanovskaya str., 197198, St. Petersburg, Russia; office phone: +7(812)347-97-70; fax: +7(812)237-12-49.

Tatarka Maxim — Ph.D., Doctoral Student, Mozhaisky Military Space Academy. Research interests: probability theory and mathematical statistics, pattern recognition, analysis of data. The number of publications — 22. maksimtbv@gmail.com; 13, Zhdanovskaya str., 197198, St. Petersburg, Russia; office phone: +7(812)347-97-70; fax: +7(812)237-12-49.

Tsyganov Andrey — Ph.D., Senior Lecturer, Mozhaisky Military Space Academy. Research interests: probability theory and mathe-matical statistics, pattern recognition, technical analysis of

signals. The number of publications — 18. porudchik@mail.ru; 13, Zdanovskaya str., 197198, St. Petersburg, Russia; office phone: +7(812)347-97-70; fax: +7(812)237-12-49.

Bessolcev Vitaliy — Lecturer, Mozhaisky Military Space Academy. Research interests: probability theory and mathematical statistics, pattern recognition, web development. The number of publications — 8. v.bessolstsev@gmail.com; 13, Zhdanovskaya str., 197198, St. Petersburg, Russia; office phone: +7(812)347-97-70; fax: +7(812)237-12-49.

References

1. Invernizzi L. et al. Cloak of visibility: detecting when machines browse a different web. Proceedings of the 37th IEEE Symposium on Security and Privacy. 2016. pp. 743–758.
2. Mowery K., Bogenreif D., Yilek S., Shacham H. Fingerprinting information in JavaScript implementations. Proceedings of W2SP. 2011. vol. 2. no. 11.
3. Pham K., Santos A.S.R., Freire J. Understanding website behavior based on user agent. Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2016. pp. 1053–1056.
4. Torres C.F., Jonker H., Mauw S. FP-Block: usable web privacy by controlling browser fingerprinting. European Symposium on Research in Computer Security. 2015. pp. 3–19.
5. Xu H., et al. Detecting and characterizing web bot traffic in a large e-commerce marketplace. European Symposium on Research in Computer Security. 2018. vol. 11099. pp. 143–163.
6. Navpreet K. et al. Browser Fingerprinting as user tracking technology. 2017 11th International Conference on Intelligent Systems and Control (ISCO). 2017. pp. 103–111.
7. Queiroz J.S., Feitosa E.L. A Web Browser Fingerprinting Method Based on the Web Audio API. *The Computer Journal*. 2019. vol. 62. no. 8. pp. 1106–1120.
8. Carpenter L. The A-buffer an Antialiased Hidden Surface Method. *IEEE Computer Graphics and applications*. 1984. vol. 18. no. 3. pp. 103–108.
9. Crow F.C. Comparison of Antialiasing Techniques. *IEEE Computer Graphics and applications*. 1981. vol. 1. no. 1. pp. 40–48.
10. Sazonov K.V., Tatarka M.V., Shuvaev F.L. [The method of structural statistical recognition of the operating modes of information and computer networks]. *Information and Space – Informaciya i Kosmos*. 2019. vol. 1. pp. 40–46. (In Russ.).
11. Sazonov K.V., Tatarka M.V., Shuvaev F.L. [The method of reducing the dimension of probabilistic suffix trees]. *Information and Space – Informaciya i Kosmos*. 2018. pp. 55–61. (In Russ.).
12. Flach P. Machine learning: the art of science of algorithms that make sense of data. Cambridge University press. 2012. 400 p.
13. Jordan M.I., Mitchell T.M. Machine learning: Trends, perspectives, and prospects. *Science*. 2015. vol. 349(6245). pp. 255–260.
14. Ayma V.A. et al. Classification algorithms for big data analysis a map reduce approach. *The International Archives of the Photogrammetry "in Remote Sensing and Spatial Information Sciences*. 2015. vol. 40. no. 3. pp 17.
15. Grolinger K. et al. Challenges for MapReduce in Big Data. IEEE 10th World Congress on Services. 2014. pp. 182–189.
16. Dhillon S., Kaur K. Comparative Study of Classification Algorithms for Web Usage Mining. *International Journal of Advanced Research in Computer Science and Software Engineering*. 2014. vol. 4(7). pp. 137–140.
17. Nikiforakis N. et al. Cookieless monster: Exploring the ecosystem of web-based device fingerprinting. 2013 IEEE Symposium on Security and Privacy (SP). 2013. pp. 541–555.

18. Khan T.M., Khan M.A.U., Kong Y. Fingerprint image enhancement using multi-scale DDFB based diffusion filters and modified Hong filters. *Optik*. 2014. vol. 125. no. 16. pp. 4206–4214.
19. Khan M.A.U., Khan T.M., Bailey D.G., Kong Y. A spatial domain scar removal strategy for fingerprint image enhancement. *Pattern Recognition*. 2016. vol. 60. pp. 258–274.
20. Khan T.M., Khan M.A.U., Kittaneh O., Kong Y. Stopping criterion for linear anisotropic image diffusion: A fingerprint image enhancement case. *EURASIP Journal on Image and Video Processing*. 2016. vol. 2016. no. 1. pp. 6.
21. Dunbar C., Qu G. A practical circuit fingerprinting method utilizing observability don't care condition. *Proceedings of the 52nd Annual Design Automation Conference*. 2015. pp. 1–6.
22. Dunbar C., Qu G. Satisfiability don't care condition based circuit fingerprinting techniques. *The 20th Asia and South Pacific Design Automation Conference (ASPDAC'15)*. 2015. pp. 815–820.