**Оглавление:**

|  |  |
| --- | --- |
| Актуальность темы | 3 |
| Постановка проблемы, гипотеза | 3 |
| Цель и задачи исследования | 4 |
| Задачи, решаемые типовыми системами видеонаблюдения | 4 |
| Анализ доступных систем и сервисов видеонаблюдения | 8 |
| Разработка программного обеспечения системы видеонаблюдения с облачным хранилищем данных | 9 |
| Исследование технических характеристик разработанной системы видеонаблюдения | 13 |
| Результаты работы | 14 |
| Выводы | 14 |
| Список литературы | 15 |
| Приложение 1. Исходный код разработанного программного обеспечения | 16 |

**Актуальность темы**

Данная тема исследования актуальна в связи с тем, что жилые и промышленные помещения, а так же прилегающая к ним территория нуждаются в постоянном контроле находящихся на них лиц и их действий. В настоящее время данная задача решается путем развертывания на охраняемом объекте систем видеонаблюдения, обладающих различным аналитическим функционалом: от простой потоковой записи архива в режиме 24/7 до высокоуровневых механизмов распознавания лиц, анализа девиантного поведения и т.д. Однако такие специализированные системы обладают значительной стоимостью - более 5 т.р., что является неприемлемым для большинства категорий физических лиц. Альтернативой дорогим специализированым системам является сервисы облачного видеонаблюдения, которые за определенную плату подключают видеокамеры клиента в свой единый интерфейс и предоставляют услуги по хранению ограниченного архива записи, а так же в ряде случаев услуги по видеоаналитике. Однако такие сервисы обладают рядом недостатков, таких как низкое качество хранимого видеоизображения для бюджетных тарифов, частая недоступность сервиса по техническим причинам (по опубликованным отзывам клиентов) и т.д. Все эти факторы приводят к актуальности разработки собственной системы видеонаблюдения, которая позволяла бы хранить видеоархив в облаке, но при этом обладала бы минимальной стоимостью для пользователя.

**Постановка проблемы**

Значительная стоимость коммерческих систем видеонаблюдения приводит к необходимости разработки собственной системы видеонаблюдения, которая позволяла бы хранить видеоархив в облаке, но при этом обладала бы минимальной стоимостью для пользователя.

**Гипотеза**

Имея в наличии обычную WEB видеокамеру и персональный компьютер с операционной системой семейства Windows можно разработать программное обеспечение для организации бюджетной системы видеонаблюдения.

**Цель исследования**.

Разработка системы видеонаблюдения, позволяющей хранить архив записи в облаке.

**Задачи:**

1. Провести анализ доступных способов организации видеонаблюдения;
2. Разработать программное обеспечение системы видеонаблюдения с облачным хранилищем данных;
3. Провести исследование технических характеристик разработанной системы видеонаблюдения.

**Задачи, решаемые типовыми системами видеонаблюдения**

Видеонаблюдение — это процесс визуального контроля (наблюдения) за объектом и происходящими на его территории событиями при помощи специализированного оборудования, объединенного в соответствующую систему. Под объектом видеонаблюдения понимается контролируемая системой видеонаблюдения территория, в том числе помещения зданий и сооружений, прилегающая территория и др. [1]. Система видеонаблюдения — это программно-аппаратный комплекс (видеокамеры, объективы, мониторы, регистраторы и др. оборудование), предназначенный для организации видеоконтроля как на локальных, так и на территориально-распределенных объектах (рисунок 1).



Рисунок 1 – Оборудование системы виденаблюдения

История появления видеонаблюдения тесно связано с телевидением, только с другими функциями и задачами, поэтому его так же называют охранным телевидением. Цель системы видеонаблюдения – обеспечение безопасности объекта (внутренних и наружных помещений, прилегающей территории и др.), людей, материальных и интеллектуальных ценностей, путем круглосуточного визуального контроля и мониторинга событий в режиме реального времени и анализа архивных данных. Основные задачи системы видеонаблюдения:

1. Обеспечение визуального контроля за объектом (в том числе внутренними и наружными помещениями и прилегающими территориями и др.).

2. Предотвращение и минимизация рисков несанкционированного проникновения и действий на подконтрольной территории.

3. Обеспечение безопасности людей, сохранности материальных и интеллектуальных ценностей.

4. Повышение уровня безопасности объекта и пользователей, за счет обеспечения дистанционного наблюдения за контрольными точками и своевременного принятия контрмер в случае возникновения необходимости без непосредственного контакта с нарушителями или опасными предметами.

5. Обеспечение своевременного информирования операторов о внештатных и опасных ситуациях.

6. Круглосуточный визуальный контроль и мониторинг состояния и событий на подконтрольной территории в режиме реального времени (в том числе за перемещением людей, предметов и др.).

7. Аккумулирование данных визуального контроля в специализированные архивы с возможностью их последующего анализа.

8. Другие, в том числе специализированные задачи, зависящие от индивидуальных характеристик систем и оборудования.

Основные функции системы видеонаблюдения:

1.Визуальное наблюдение за подконтрольной территорией и передача данных в специализированные архивы и на удаленные расстояния по каналам связи (Wi-Fi, Ethernet и др.).

2. Дистанционный контроль и мониторинг состояния объекта, в том числе на взаимодействие оборудования системы со специализированным программным обеспечением, аналитическими приложениями и встроенными аналитическими функциями (например, анализ движения и перемещения объектов, включение/выключение записи при начале движения на подконтрольной территории, автоматическое приближение/удаление объекта, работа по сценариям и др.).

3. Интеграция с другими системами, например, охрана периметра, системой распознавания авто номеров, учета парковочных мест, учета рабочего времени и др. системами;

4. Запись и формирование систематизированного архива событий;

5. Круглосуточный визуальный контроль и мониторинг состояния объекта, отслеживание нестандартных, нештатных и опасных ситуаций, своевременное информирование о них операторов, персонала и посетителей.

6. Анализ данных, формирование отчетов и др.

7. Другие функции, в том числе специализированные и зависящие от индивидуального назначения системы и характеристик оборудования.

Области применения систем видеонаблюдения:

1. На транспортных объектах – автобусы, метро, электрички, поезда, такси, маршрутные такси, личный транспорт и др. Организуется с целью обеспечения в режиме реального времени визуального контроля за действиями лица, управляющего транспортным средством, пассажиров и других участников движения. Минимизации, предотвращения и оперативного пресечения рисков совершения правонарушений, нестандартных и внештатных ситуаций, фиксирования (записи в архив) событий и действий, оперативного и последующего анализа событий, учета рабочего времени персонала, пиковых нагрузок транспорта и др. аналитических и прочих функций;

2. На муниципальных и транспортных объектах – вокзалы, платформы метро и электричек, аэропорты, школы, детские сады и др. объекты. Видеонаблюдение организуется с целью обеспечения безопасности персонала, посетителей и имущества, визуального контроля событий в режиме реального времени и их фиксации в архиве. Оперативного мониторинга, предотвращения, оперативного пресечения и последующего анализа всех возможных ситуаций и рисков террористического характера, правонарушений, опасности для жизни и имущества государственной, муниципальной, личной собственности и др. Дополнительно может вестись мониторинг и учет рабочего времени, посещаемости, режима нагрузок объекта и взаимодействие с другими системами.

3. На развлекательных и торговых объектах - торговые центры, кинотеатры, рестораны и др. Организуется для обеспечения общей безопасности объекта, персонала, имущества, посетителей, а так же для выявления и отслеживания действий недобросовестного персонала, мошенников, мониторинга и пресечение случаев воровства со стороны посетителей и персонала, контроля трудовой дисциплины и учета рабочего времени персонала, отслеживание и пресечение случаев нарушений в товарообороте, сбоев логистических систем (задержки поступления, разгрузки товара), отслеживание и анализ действий персонала, выявление нарушений, оптимизация работы.

4. Также системы видеонаблюдения используются для контроля за офисными помещениями, где может исходить угроза сотрудникам, всевозможному инвентарю, оргтехнике и даже интеллектуальной собственности. При их помощи можно осуществлять контроль за соблюдением правил техники безопасности, что позволяет значительно уменьшить травматизм, а также летальные случаи на производстве. Они сочетаются с системами контроля рабочего времени, всевозможными пожарными и охранными системами.

5. Частный сектор – квартиры, дачи, коттеджи, гаражи и др. – обеспечение безопасности людей и имущества, круглосуточного визуального контроля за состоянием объекта, фиксации данных, организации удаленного контроля и доступа, минимизации и своевременного пресечения рисков несанкционированного проникновения, совершения противоправных действий и др.

6. На особо охраняемых и секретных объектах – для обеспечения безопасности персонала и посетителей, сохранности имущества, оборудования, интеллектуальной собственности, обеспечения трудовой дисциплины, мониторинга и контроля соблюдения правил и требований безопасности, протоколов работы с оборудованием и информацией и др. Круглосуточного визуального контроля за событиями, перемещениями людей, предметов, техники и оборудования, анализа событий в режиме реального времени и архивных данных и др.

**Анализ доступных систем и сервисов видеонаблюдения**

В настоящее время на рынке доступно большое разнообразие готовых к развертыванию автономных систем видеонаблюдения [2-4], состоящих из набора видеокамер и регистратора. Недостатком таких систем является высокая цена и отсутствие облачного хранилища – архив хранится на регистраторе и может быть уничтожен злоумышленниками, проникшими на охраняемый объект.

Помимо этого, активно развиваются сервисы облачного видеонаблюдения [5], включающие видеокамеры пользователя в свой единый интерфейс. Однако качество функционирования этих сервисов оставляет желать лучшего (по отзывам пользователей), при том что абонентская плата за пользование сервисом в месяц начинается от 200 рублей за одну подключенную камеру, что зачастую является неприемлемым для физических лиц (таблица 1).

Такое положение дел приводит к необходимости разработки собственной системы видеонаблюдения, не требующей абонентской платы за свое использование. Дополнительным требованием к такой системе является подключение к облачному хранилищу данных в режиме бесплатного использования.

*Таблица 1. Облачные сервисы видеонаблюдения*

|  |  |  |
| --- | --- | --- |
| Наименование | Тариф за одну камеру | Источник |
| [[LinkVideo](https://crmindex.ru/services/linkvideo)](https://crmindex.ru/services/linkvideo) | 250 руб/месяц | https://linkvideo.ru/?utm\_source=crmindex |
| [YOULOOK](https://crmindex.ru/services/youlook) | 250 руб/месяц | https://youlook.ru/ |
| [NOVIcloud](https://crmindex.ru/services/novicloud) | 200 руб/месяц | http://cloud.novicam.ru/index |
| [IPEYE](https://crmindex.ru/services/ipeye) | 3.3 руб/сутки | https://www.ipeye.ru/ |
| [CamDrive](https://crmindex.ru/services/camdrive) | Данные отсутствуют | https://www.camdrive.ru/ |
| [EZVIZ](https://crmindex.ru/services/ezviz) | Данные отсутствуют | https://ezviz.ru/ |
| [Линия](https://crmindex.ru/services/liniya) | 7 руб/сутки | https://devline.ru/cloud\_server/ |
| [I-CAM](https://crmindex.ru/services/icam) | 270 руб/ месяц | http://i-cam.ru/ |
| [WebGlazok](https://crmindex.ru/services/webglazok) | 179 руб/месяц | https://webglazok.com/ |
| [IVideon](https://crmindex.ru/services/ivideon) | Данные отсутствуют | https://ru.ivideon.com/ |

**Разработка программного обеспечения системы видеонаблюдения с облачным хранилищем данных**

Схема макета разрабатываемой системы видеонаблюдения представлена на рисунке 2. В соответствии со схемой WEB камера подключается через USB порт к компьютеру под управлением ОС Windows. Разработанное в рамках проекта программное обеспечение **WebCapToFiles** осуществляет подключение к камере, получение от нее видеопотока, его кодирование (сжатие) и запись сжатого потока в медиафайл в режиме «нарезки» файлов по времени (по истечении заданного времени текущий файл закрывается и открывается новый) в заданный каталог. Приложение WebCapToFiles при запуске принимает в командной строке два числовых параметра: периодичность «нарезки» выходных файлов в секундах и частоту кадров для заголовка выходного видеофайла (обычно выставляется в 25);



Рисунок 2 – Макет разработанной системы видеонаблюдения

В системе запущена стороння утилита **yandex-disk-upload** сканирования каталога и выгрузки, появляющихся в нем файлов на облачный диск Yandex-Drive с заданной адресной информацией [6]. Стоит заметить, что утилита разработана сторонними специалистами и не является частью защищаемого проекта. Размер бесплатного хранилища Yandex-Drive в настоящее время составляет 10 Гбайт, что позволяет хранить на нем приемлемый объем данных (рисунок 3). Утилита yandex-disk-upload осуществляет циклическую запись файлов на удаленном диске – то есть при нехватке места удаляется самый старый файл.

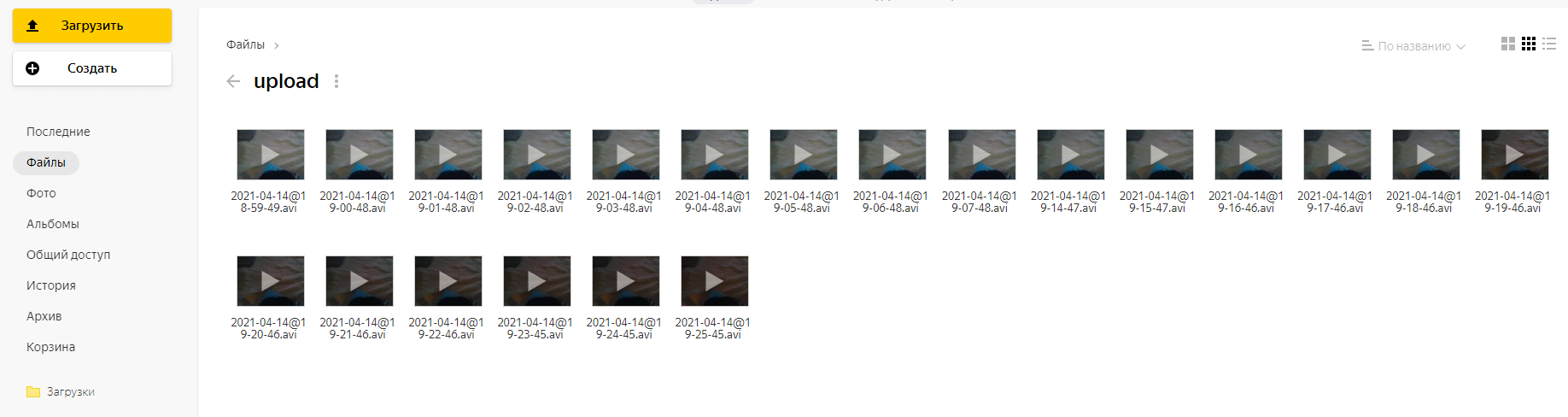


Рисунок 3 – Личный кабинет на Yandex Drive с доступным архивом записей

Таким образом весь процесс функционирования макета выглядит следующим образом:

1. Приложение WebCapToFiles осуществляет получение видеопотока от камеры и его запись в файлы по заданному времени в заданный каталог.
2. Утилита yandex-disk-upload осуществляет сканирование заданного каталога и при обнаружении новых файлов производит их выгрузку в облачное хранилище Yandex-Drive с циклической перезаписью.
3. После выгрузки в облачное хранилище исходные файлы на компьютере удаляются.
4. В облачном хранилище Yandex-Drive всегда доступен архив записи камеры с какой-то ограниченной историей. В имени файла содержится временная метка, позволяющая быстро найти требуемый временной промежуток записи.

Исходный код разработанного в ходе выполнения проекта программного обеспечения WebCapToFiles представлен в Приложении 1. Приложение осуществляет подключение к USB камере через штатный интерфейс программного пакета OPENCV. Затем выполняется открытие текущего выходного файла и запись в него видеоданных. По истечении заданного временного промежутка текущий файл закрывается и на запись открывается новый файл.

Состав пакета программного обеспечения разработанной системы видеонаблюдения представлен в таблице 2.

*Таблица 2. Состав пакета программного обеспечения разработанной системы видеонаблюдения*

|  |  |
| --- | --- |
| Файл | Назначение |
| WebCapToFiles.exe | Основное приложение системы. Осуществляет подключение в WEB камере и запись видеопотока в файлы в режиме «нарезки» |
| opencv\_world450.dll | Служебная библиотека для взаимодействия с пакетом OPENCV |
| opencv\_videoio\_ffmpeg450\_64.dll | Служебная библиотека для взаимодействия с пакетом OPENCV |
| yandex-disk-upload.exe | Стороння утилита выгрузки файлов на удаленный Yandex Drive |
| yandex-disk-upload.conf | Конфигурационный файл утилиты yandex-disk-upload.exe |

Структура конфигурационного файла утилиты yandex-disk-upload.exe представлена на рисунке 4. Назначение параметров доступно по ссылке [6] и раскрыто в таблице 3.

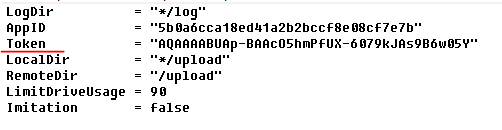


Рисунок 4 – Структура конфигурационного файла утилиты yandex-disk-upload.exe

*Таблица 3. Назначение параметров конфигурационного файла*

*yandex-disk-upload.conf*

|  |  |
| --- | --- |
| Поле | Назначение |
| LogDir | Путь к директории LOG файлов |
| AppID | Глобальный идентификатор утилиты yandex-disk-upload.conf, выданный системой Yandex |
| Token | Локальный идентификатор, выданный конкретным акаунтом Yandex для взаимодействия с утилитой |
| LocalDir | Локальная директория для сканирования и поиска файлов для выгрузки |
| RemoteDir | Имя директории в корневом каталоге Yandex для записи файлов |
| LimitDriveUsage | Ограничение общего размера хранимых данных в процентах от доступного объема, после переполнения которого старые файлы будут удаляться (режим циклической перезаписи) |

**Исследование технических характеристик разработанной системы видеонаблюдения**

Разработанная система видеонаблюдения для 1 минуты записи формирует файл размером около 15 Мбайт. Размер бесплатного хранилища Yandex Drive составляет 10 Гбайт. Следовательно, в удаленном хранилище Yandex можно бесплатно хранить архив длительностью 10\*1024/15=682 минуты (11 часов) видео.

При этом, минимальный тариф от Yandex по подписке составляет 99 р/мес (990 р/год) за 100 Гбайт [7], что позволяет организовать хранение 100\*1024/15=6826 минут (113 часов, 4 суток) непрерывной записи. Сводная информация о достижимом объеме архива записей представлена в таблице 4.

*Таблица 4. Сводная информация об объемах архива записей*

|  |  |  |
| --- | --- | --- |
| Тариф Yandex | Размер абонентской платы (р/год) | Размер архива |
| Бесплатный | 0 | 11 часов |
| Стандарт 100 ГБ | 990 | 4 суток |

**Результаты работы:**

На основе анализа доступных систем видеонаблюдения принято решение в рамках проекта разработать собственную систему, требующую минимальных финансовых затрат на свое функционирование и хранящую архив записей в облачном хранилище. Разработанное программное обеспечение функционирует под управлением операционной системы семейства Windows и позволяет хранить архив записи от 11 часов до 4 суток.

**Выводы:**

1. Современные системы видеонаблюдения обладают высокой стоимостью, или высокой абонентской платой за использование сервиса, что не позволяет физическим лицам приобретать их в массовом порядке. Кроме того, функционирование таких систем вызывает серьезные нарекания пользователей.
2. В рамках проекта разработана собственная система видеонаблюдения, функционирующая при минимальных финансовых затратах. Хранение архива записей осуществляется на удаленном Yandex диске.
3. Проведен выбор параметров функционирования системы, позволивший обеспечить хранение данных в архиве от 11 часов (бесплатно) до 4 суток (на минимальном тарифе от Yandex).
4. В дальнейшем планируется осуществить портирование системы на ОС семейства Linux и ее запуск на бюджетных одноплатных компьютерах типа Orange-PI [8].
5. Кроме того, программный пакет OPENCV позволят проводить широкий ряд аналитических операций по обработке видеопотока: детектирование объектов, анализ движения. Планируется так же изучить эти возможности.

**Список литературы:**

* 1. Бадаев В.А. Разработка, внедрение и эксплуатация систем <https://dspace.tltsu.ru/bitstream/123456789/2639/1/%D0%91%D0%B0%D0%B4%D0%B0%D0%B5%D0%B2%20%D0%92.%D0%90._%D0%AD%D0%9B%D0%BC-1401.pdf>
  2. <https://rusmarta.ru/market/komplekty_videonablyudeniya/>
  3. <https://carcam.ru/video-monitoring/video-kit/>
  4. <https://aliexpress.ru/popular/cctv-surveillance-kit.html>
  5. <https://crmindex.ru/ratings/servisy_oblachnogo_videonabludeniya?page=2>
  6. <https://github.com/msw-x/yandex-disk-upload>
  7. https://mail360.yandex.ru/?from=disk\_buybtn
  8. <http://www.orangepi.org/>

Приложение 1.

Исходный код разработанного программного обеспечения

|  |
| --- |
| #pragma once  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <iostream>  #include <filesystem>  #include "opencv2/core/utility.hpp"  #include "opencv2/video/tracking.hpp"  #include "opencv2/imgproc.hpp"  #include "opencv2/videoio.hpp"  #include "opencv2/highgui.hpp"  #include <windows.h>  using namespace cv;  using namespace std;  const string currentDateTime()  {  // строка-метка времени для имени файла  time\_t now = time(0);  struct tm tstruct;  char buf[80];  tstruct = \*localtime(&now);  strftime(buf, sizeof(buf), "%Y-%m-%d@%H-%M-%S", &tstruct);  return buf;  }  auto start = chrono::system\_clock::now(); // инициализация значения времени  bool GetTimeEvent(int seconds)  {  // поиск события - превышение времени  auto end = chrono::system\_clock::now();  chrono::duration<double> elapsed\_seconds = end - start;  if (elapsed\_seconds.count() > seconds) {  start = chrono::system\_clock::now(); // обновление стартового значения времени  return 1;  }  return 0;  }  void OpenVideoFile(VideoWriter& video, int codec, int frame\_width, int frame\_height, string video\_directory, int fps)  {  // открытие видеофоайла на запись  string const basename = currentDateTime() + ".avi";  cout << "write file: " << basename << endl;  string const filename = video\_directory + basename;  video.open(filename, codec, fps, Size(frame\_width, frame\_height), true);  }  int main(int ac, char \*av[]) {  VideoCapture cap;  int camNum = 0; // номер видеопотока  bool paused = false;  cap.open(camNum);  Mat frame;    // задание выходной директории (текущая + \video\)  string const video\_directory = filesystem::current\_path().string() + "\\video\\";  cout << "write files to directory: " << video\_directory << endl;  ::CreateDirectoryA(video\_directory.c\_str(), NULL);  // gпараметры видео от камеры  int codec = VideoWriter::fourcc('X', 'V', 'I', 'D'); //cap.get(CAP\_PROP\_FOURCC);  int frame\_width = cap.get(CAP\_PROP\_FRAME\_WIDTH);  int frame\_height = cap.get(CAP\_PROP\_FRAME\_HEIGHT);  // получение данных из командной строки  int fps = 20;  int durationOneFileSeconds = 60; // количество секунд видео в одном файле  if (ac > 1) fps = stoi(av[1]);  if (ac > 2) durationOneFileSeconds = stoi(av[2]);  cout << "rate: " << fps << " fps" << endl;  cout << "limit: " << durationOneFileSeconds << " sec" << endl;  //проверка корректности подключения к камере  if (!cap.isOpened())  {  cout << "could not initialize capturing" << endl;  return 0;  }  namedWindow("Cam", 0);  // открытие файла на запись  VideoWriter video;  OpenVideoFile(video, codec, frame\_width, frame\_height, video\_directory, fps);  while (true) {  if (!paused) {  cap >> frame;  if (frame.empty())  break;  // переоткрытие нового файлв\а на запись  if (GetTimeEvent(durationOneFileSeconds)) {  video.release();  OpenVideoFile(video, codec, frame\_width, frame\_height, video\_directory, fps);  }  // запись очередного кадра в файл  video.write(frame);  }  // отображение очередного кадра  imshow("Cam", frame);  // Обработка нажатия клавиш  char c = (char)waitKey(10);  if (c == 27)  break;  switch (c)  {  case 'p':  paused = !paused;  break;  default:  ;  }  }  return 0;  } |