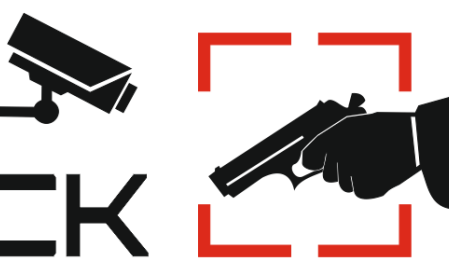
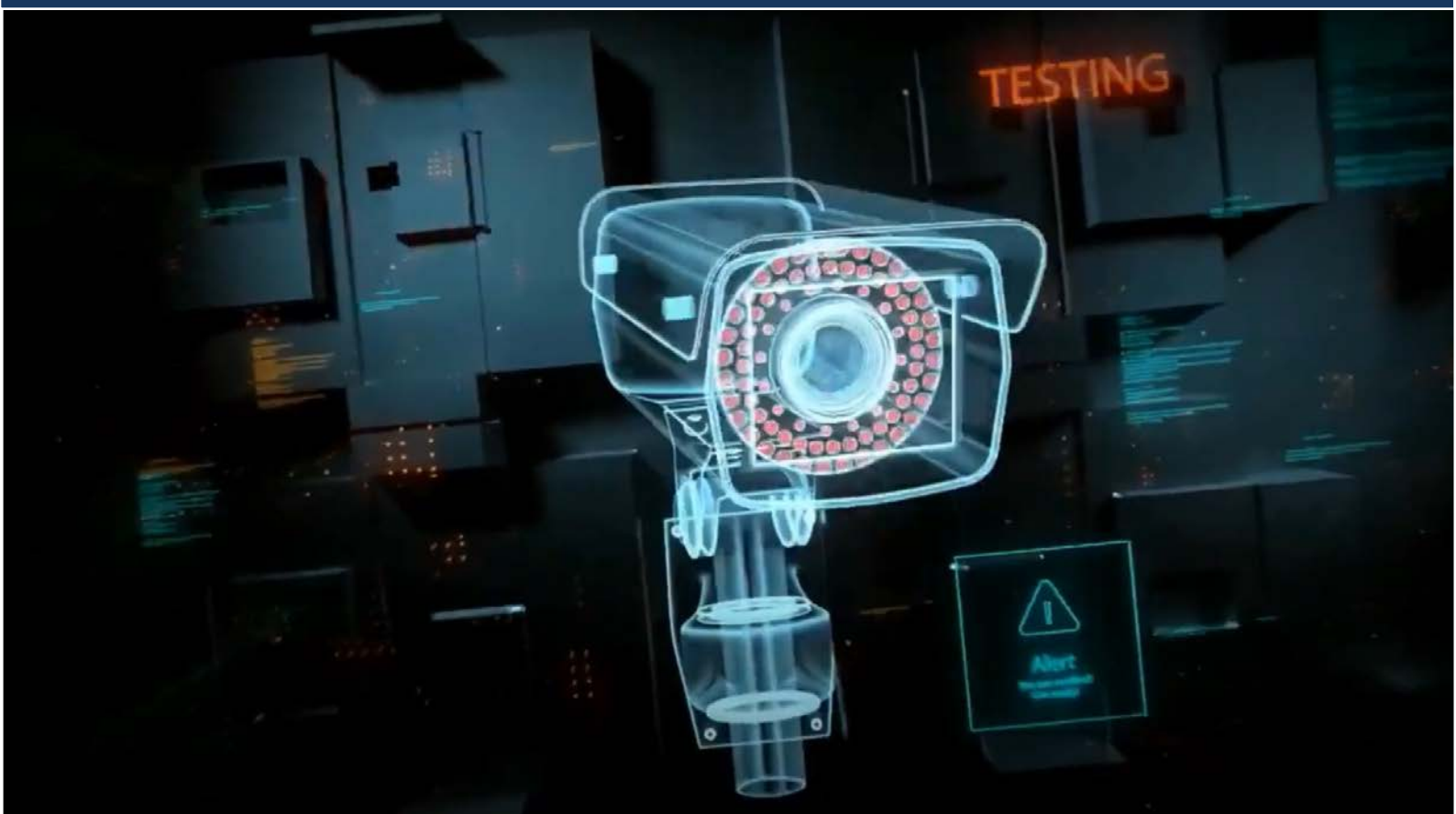


GUN
CHECK



Программа для распознавания
оружия с камер видеонаблюдения в
режиме реального времени при
помощи искусственного интеллекта



metakom.tech

METAKOM-ALPHA - 2022

Оглавление

Описание программы.....	2
Характеристики	2
Скорость работы.....	2
Серверный режим	2
Технические требования.....	3
Минимальные требования для рабочего места:.....	3
Оптимальные требования для рабочего места:	3
Требования к камерам:.....	3
Требования к операционной системе.....	4
Подготовка и установка демо-версии ПО «GunCheck»	4
Ограничения демо-версии ПО «GunCheck».....	4
Установка зависимостей демо-версии программы «GunCheck» для ОС Ubuntu 20.04.	4
Установка демо-версии ПО «GunCheck».....	5
Подготовка к работе демо-версии	5
Подготовка и установка ПО «GunCheck»	5
Установка зависимостей программы «GunCheck» для ОС Ubuntu 20.04.	5
Очистка системы от установленных ранее драйверов и пакетов	6
Скачивание локальных версий пакетов.....	6
Установка OpenCV-4.6.0 с зависимостями и поддержкой CUDA и CUDNN.....	7
Установка поддержки звука.....	9
Установка поддержки транскодирования видео (опционально)	9
Установка поддержки SQL БД (опционально)	9
Установка версии ПО «GunCheck»	11
Подготовка к работе	11
Описание интерфейса пользователя	13
Главный экран.....	13

Описание программы

Программа «**GunCheck**» позволяет мгновенно распознавать вооруженных людей, как в потоке людей в городских условиях, так и на открытой местности при помощи камер видеонаблюдения, которые сегодня используются повсеместно.

После определения угрозы, программа обратит внимание оператора на нужную камеру и выполнит запрограммированные действия.

Характеристики

Скорость работы

На бюджетном ГПУ (Mobile RTX 3070 с 8 GB памяти) поддерживается до 4 камер с высоким FPS, с возможностью распознавания оружия на дистанциях до 30 метров.

Если снизить расстояние до 15 метров, то система обеспечит работу уже 16 камер с аналогичным числом кадров в секунду. При использовании камер с кодеком H.265 можно поднять КПД ещё выше, за счёт пропуска обработки кадров, в которых нет изменений в изображении

Серверный режим

Существует возможность работы в серверном режиме и подбор оборудования под ваши нужды и возможности, в том числе использование имеющегося.

Технические требования

Минимальные требования для рабочего места:

Центральный процессор (CPU)	8 ядер с частотой 2 ГГц
Оперативная память (RAM)	8 Гб
Видеокарта (CUDA)	RTX 2060, 8 Гб

Оптимальные требования для рабочего места:

Центральный процессор (CPU)	8 ядер с частотой 2 ГГц
Оперативная память (RAM)	16 Гб
Видеокарта (CUDA)	RTX 3070, 8 Гб

Требования к камерам:

Видеокодек	H.265 / H.265+
Поддержка протокола	RTSP
Разрешение матрицы (Megapixel)	4-5 MP
Скорость видеопотока (Bitrate)	8-16 MB/s
Разрешение выходного потока (Pixels)	1920 x 1080
Количество кадров в секунду (FPS)	25

Требования к операционной системе

Операционная система должна удовлетворять следующим требованиям:

- Linux/GTK
- Bash
- C++ (кроме демо-версии программы)
- CUDA (кроме демо-версии программы)
- OpenCV (для демо-версии устанавливается автоматически, для остальных версий требуется ручная установка)

Подготовка и установка демо-версии ПО «GunCheck»

Ограничения демо-версии ПО «GunCheck»

Демо-версия ПО «GunCheck» предоставляется исключительно для оценки пользователем интерфейса программы и не предназначена для распознавания оружия в реальных проектах. Используемая в демо-версии нейросеть урезана по своим рабочим параметрам до возможности работать только на CPU компьютера без использования GPU, обязательного для клиентских и серверных версий ПО «GunCheck».

Искусственно ограничены следующие параметры нейросети:

- дальность распознавания оружия
- качество распознавания оружия
- распознавание короткоствольного оружия

Установка зависимостей демо-версии программы «GunCheck» для ОС Ubuntu 20.04.

Убедитесь, что в файле `/etc/apt/sources.list` указан репозиторий `archive.ubuntu.com` со следующими параметрами:

```
deb http://archive.ubuntu.com/ubuntu/ focal-updates main restricted universe multiverse
```

если он отсутствует, то добавьте его первым по списку репозиторию.

Обновите установленные пакеты до текущих версий:

```
sudo apt update  
sudo apt upgrade
```

Установка демо-версии ПО «GunCheck»

Скопируйте файл **gunccheck-demo-1.0_1.deb** в выбранную папку и выполните команду:

```
sudo apt install ./gunccheck-demo-1.0_1.deb
```

Подготовка к работе демо-версии

Для описания видеоисточника, достаточно указать параметры соединения с камерами.

В командной строке формат параметров:

```
--urlN=url
```

Для предоставленной демо-версии клиентского комплекса ПО N может быть в пределах от 1 до 4.

Пример:

```
gunccheck ---url1=rtsp://192.168.1.168/onvif1 |  
--url2=rtsp://login:password@192.168.1.169:5554/onvif1 |  
--url3=rtsp://login:password@192.168.1.170 |  
--url4=rtsp://login:password@192.168.1.171/ch01/0
```

Далее, если вы пользуетесь демо-версией ПО «GunCheck», можно переходить к инструкции пользователя клиентской версии.

Подготовка и установка ПО «GunCheck»

Установка зависимостей программы «GunCheck» для ОС Ubuntu 20.04.

Убедитесь, что в файле `/etc/apt/sources.list` указан репозиторий `archive.ubuntu.com` со следующими параметрами:

```
deb http://archive.ubuntu.com/ubuntu/ focal-updates main restricted universe multiverse
```

если он отсутствует, то добавьте его первым по списку репозиториум.

Обновите установленные пакеты до текущих версий:

```
sudo apt update  
sudo apt upgrade
```

Для установки полноценной версии ПО «GunCheck» требуется самостоятельная установка ПО CUDA/CuDNN/TensorRT с сайта NVidia и настройка компиляции OpenCV с указанием доступности этих пакетов.

Очистка системы от установленных ранее драйверов и пакетов

```
sudo apt-get --purge remove nvidia  
sudo apt-get --purge remove "*nvidia*"  
sudo apt-get --purge remove "*cublas*" "cuda*" "nsight*"  
sudo rm -rf /usr/local/cuda*  
sudo apt-get autoremove  
sudo apt-get autoclean  
sudo reboot
```

Скачивание локальных версий пакетов

- *cuda-repo-ubuntu2004-11-6-local_11.6.2-510.47.03-1_amd64.deb*
- *cudnn-local-repo-ubuntu2004-8.4.0.27_1.0-1_amd64.deb*
- *nv-tensorrt-repo-ubuntu2004-cuda11.6-trt8.4.0.6-ea-20220212_1-1_amd64.deb*

в данной инструкции не рассматривается. Для их получения необходимо перейти на сайт: <https://developer.nvidia.com>. До скачивания уточните в разделе документации, какие версии пакетов CUDA, CUDNN и TensorRT совместимы между собой в настоящий момент.

Подробная инструкция расположена на сайте NVidia: <https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>

Так же следует учитывать, что у вас может быть устаревшее оборудование (например, NVidia K80), поддержка которого может отсутствовать в текущих версиях пакетов. В этом случае необходимо будет найти и скачать их архивные версии.

Ниже показан пример одной из возможных последовательностей команд, ведущих к установке пакетов CUDA, CUDNN и TensorRT:

```
sudo apt-get install nvidia-driver-510  
sudo reboot
```

```

wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin
sudo mv cuda-ubuntu2004.pin /etc/apt/preferences.d/cuda-repository-pin-600
sudo dpkg -i cuda-repo-ubuntu2004-11-6-local_11.6.2-510.47.03-1_amd64.deb
sudo apt-key add /var/cuda-repo-ubuntu2004-11-6-local/7fa2af80.pub
sudo apt-get update
sudo apt-get -y install cuda
sudo dpkg -i cudnn-local-repo-ubuntu2004-8.4.0.27_1.0-1_amd64.deb
sudo apt-get update
sudo apt-get install libcudnn8=8.4.0.27-1+cuda11.6
sudo apt-get install libcudnn8-dev=8.4.0.27-1+cuda11.6
sudo dpkg -i nv-tensorrt-repo-ubuntu2004-cuda11.6-trt8.4.0.6-ea-20220212_1-1_amd64.deb
sudo apt-get update
sudo apt-get install tensorrt
sudo apt-get install python3-libnvinfer-dev
sudo -H pip3 install -U pip numpy onnx
sudo apt-get install onnx-graphsurgeon

```

Установка OpenCV-4.6.0 с зависимостями и поддержкой CUDA и CUDNN

Далее предполагается, что архивы opencv.zip и opencv_contrib.zip будут скачаны в каталог ~/Downloads/, что явно учитывается в переменной OPENCV_EXTRA_MODULES_PATH во время сборки OpenCV утилитой cmake.

```

cd ~
mkdir Downloads
cd Downloads
wget -O opencv.zip https://github.com/opencv/opencv/archive/refs/tags/4.6.0.zip
wget -O opencv_contrib.zip https://github.com/opencv/opencv_contrib/archive/refs/tags/4.6.0.zip
sudo apt-get install libnetplan0
sudo apt-get install netplan.io
sudo apt-get install libyaml-cpp-dev
sudo apt-get install curl
sudo apt-get install libcurl4-openssl-dev
sudo apt install build-essential cmake pkg-config unzip yasm git checkinstall
sudo libjpeg-dev libpng-dev libtiff-dev
sudo apt install libavcodec-dev libavformat-dev libswscale-dev libavresample-dev
sudo apt install libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev

```

```

sudo apt install libxvidcore-dev x264 libx264-dev libfaac-dev libmp3lame-dev libtheora-dev
sudo apt install libfaac-dev libmp3lame-dev libvorbis-dev
sudo apt install libopencore-amrnb-dev libopencore-amrwb-dev

sudo apt-get install libdc1394-22 libdc1394-22-dev libxine2-dev libv4l-dev v4l-utils
pushd /usr/include/linux
sudo ln -s -f ../libv4l1-videodev.h videodev.h
popd
sudo apt-get install python3-dev python3-pip
sudo apt install python3-testresources
sudo apt-get install libgtk-3-dev
sudo apt-get install libtbb-dev
sudo apt-get install libatlas-base-dev gfortran
sudo apt-get install libprotobuf-dev protobuf-compiler
sudo apt-get install libgoogle-glog-dev libgflags-dev
sudo apt-get install libgphoto2-dev libeigen3-dev libhdf5-dev doxygen
sudo apt-get install git
unzip opencv_contrib.zip
unzip opencv.zip
cd opencv-4.6.0
mkdir build
cd build

```

Для сборки необходимо указать правильную версию архитектуры CUDA SM в переменной `CUDA_ARCH_BIN`. Для вашего GPU вы можете посмотреть её на сайте NVidia: <https://developer.nvidia.com/cuda-gpus> или кратко, по поколениям: <https://docs.nvidia.com/cuda/cuda-compiler-driver-nvcc/index.html#virtual-architecture-feature-list>

```

cmake -D CMAKE_BUILD_TYPE=RELEASE |
-D CMAKE_INSTALL_PREFIX=/usr/local |
-D WITH_TBB=ON |
-D ENABLE_FAST_MATH=1 |
-D CUDA_FAST_MATH=1 |
-D WITH_CUBLAS=1 |
-D WITH_CUDA=ON |
-D BUILD_opencv_cudacodec=OFF |
-D WITH_CUDNN=ON |
-D OPENCV_DNN_CUDA=ON |

```

```

-D CUDA_ARCH_BIN=8.6 |
-D WITH_V4L=ON |
-D WITH_QT=OFF |
-D WITH_OPENGL=ON |
-D WITH_GSTREAMER=ON |
-D OPENCV_GENERATE_PKGCONFIG=ON |
-D OPENCV_PC_FILE_NAME=opencv.pc |
-D OPENCV_ENABLE_NONFREE=ON |
-D OPENCV_PYTHON3_INSTALL_PATH=/usr/lib/python3/dist-packages |
-D PYTHON_EXECUTABLE=/usr/bin/python3 |
-D OPENCV_EXTRA_MODULES_PATH=~/.Downloads/opencv_contrib-4.6.0/modules |
-D INSTALL_PYTHON_EXAMPLES=OFF |
-D INSTALL_C_EXAMPLES=OFF |
-D BUILD_PERF_TESTS=OFF |
-D BUILD_TESTS=OFF |
-D BUILD_EXAMPLES=OFF ..
make
sudo make install

```

Установка поддержки звука

```

sudo apt-get install alsa-base alsa-utils gstreamer1.0-alsa
sudo apt-get install libasound2 libasound2-data libasound2-plugins
sudo apt-get install libasound2-dev linux-sound-base

```

Установка поддержки транскодирования видео (опционально)

```

sudo apt-get install gstreamer1.0-plugins-base gstreamer1.0-plugins-good
sudo apt-get install gstreamer1.0-plugins-ugly
sudo apt-get install gstreamer1.0-rtsp gstreamer1.0-tools

```

Установка поддержки SQL БД (опционально)

В случае приобретения ПО «GunCheck» с поддержкой работы с SQL БД, вы можете использовать БД для хранения истории сработок, обеспечения доступа к ним через web-интерфейс посредством REST-запросов и поддержки пользовательского механизма нотификации (реализован для PostgreSQL и Oracle).

ПО «GunCheck» работает с БД поддерживающими ODBC-драйвера. Версии с поддержкой Oracle OCI или других высокопроизводительных интерфейсов к БД предоставляются по запросу.

В БД должна присутствовать таблица `alarms`. В случае использования PostgreSQL, она может быть создана с использованием следующего DDL:

```
CREATE TABLE alarms (
    id serial primary key,
    cam int not null,
    tstamp timestamp not null,
    rec_class int not null,
    img bytea not null
);
```

Если вы планируете использовать механизм поддержки пользовательской нотификации, с нашей реализацией в качестве основы, вам необходима очередь сообщений с именем `new_alarm_added`. В случае использования PostgreSQL, она может быть создана с использованием следующего DDL:

```
CREATE or REPLACE FUNCTION notify_new_alarm()
    RETURNS trigger
    LANGUAGE 'plpgsql'
as $BODY$
declare
begin
    if (tg_op = 'INSERT') then

        perform pg_notify('new_alarm_added',
            json_build_object(
                'id', NEW.id,
                'cam', NEW.cam,
                'tstamp', NEW.tstamp,
                'rec_class', NEW.rec_class
            )::text);
    end if;

    return null;
end
$BODY$;
```

```
CREATE TRIGGER after_insert_alarm
```

```
AFTER INSERT
ON alarms
FOR EACH ROW
EXECUTE PROCEDURE notify_new_alarm();
```

Установка версии ПО «GunCheck»

Скопируйте файл **gunccheck-1.0_1.deb** в выбранную папку и выполните команду:

```
sudo apt install ./gunccheck-1.0_1.deb
```

Подготовка к работе

Все аргументы командной строки, кроме имени нейросети, задаются в формате — *key=value*

Обязательными параметрами для запуска ПО «GunCheck» являются только описания видеоисточников. Для описания видеоисточника, достаточно указать параметры соединения с камерами.

В командной строке формат параметров:

```
--urlN=url для сетевого видеоисточника или
--inputN=path для использования видеофайла
```

Для демо-комплекса ПО, передаваемого с целью оценки пользователем интерфейса программы, N может быть в пределах от 1 до 4.

Для клиентского комплекса ПО N может быть в пределах от 1 до 4...64, в зависимости от версии программы и договора поставки.

Для серверной версии ПО N может быть в пределах от 1 до 4...1024, в зависимости от версии программы и договора поставки.

Пример:

```
gunccheck --url1=rtsp://192.168.1.168/onvif1 |
--url2=rtsp://login:password@192.168.1.169:5554/onvif1 |
--url3=rtsp://login:password@192.168.1.170 |
--url4=rtsp://login:password@192.168.1.171/ch01/0
```

В случае, если договор поставки подразумевает версию ПО с несколькими нейросетями, дополнительно в командной строке должно быть указано имя нейросети:

```
gunccheck ---url1=rtsp://192.168.1.168/onvif1 |  
--url2=rtsp://login:password@192.168.1.169:5554/onvif1 |  
--url3=rtsp://login:password@192.168.1.170 |  
--url4=rtsp://login:password@192.168.1.171/ch01/0 |  
HighResNN
```

Дополнительно, для клиентской и серверной версии ПО «GunCheck» доступны следующие параметры командной строки:

--thr Уровень тревожности нейросети. По умолчанию 0.3.

--ui Выбор фронтэнда. Возможные варианты: *none* для работы без GUI, *sciter* для работы с Sciter UI, *opencv* тестовый режим для отладки и разработки, не предназначен для использования клиентами.

--alarm-path Путь к каталогу, где будут храниться кадры сработок (в случае хранения сработок в файловой системе).

--sql-url Параметры соединения с SQL БД, где будут храниться кадры сработок (в случае хранения сработок в SQL БД).

--ceph-url Параметры соединения с Ceph-хранилищем (в случае хранения сработок в Ceph).

--fs-alarms разрешение использовать файловую систему для хранения сработок.

--sql-alarms разрешение использовать SQL БД для хранения сработок.

--ceph-alarms разрешение использовать Ceph-хранилище для хранения сработок.

--alarms-rate Ограничение скорости записи сработок в SQL БД, Ceph или файловую систему. Задаётся в миллисекундах. Например, *--alarms-rate=100*, означает записывать не более 10 сработок в секунду.

--bot-id Позволяет указать идентификатор бота, для отправки сообщений Telegram-чат.

--bot-chat Позволяет указать идентификатор чата, для отправки сообщений Telegram-чат.

--delay Позволяет ограничить скорость обработки кадров архивных видеопотоков ускорителем, задаётся в миллисекундах на кадр.

--out-file Позволяет записать распознанный видеопоток с метками сработок в файл.

--out-url Позволяет записать распознанный видеопоток с метками сработок в сетевой видеопоток по протоколу RTSP.

--out-fps Позволяет задать желаемую частоту кадров для выходного видеопотока.

--scada-delay В случае интеграции со СКУД, позволяет задать задержку перед сработкой сценария СКУД. Значение по умолчанию 30 секунд.

--scada-port В случае интеграции со СКУД, порт для передачи команды на запуск сценария в СКУД.

--sms В случае интеграции с IP-телефонией или GSM-шлюзом, разрешение посылать SMS сообщение о сработке на указанный номер или номера.

--sound Путь к файлу, который будет проигрываться в случае сработки нейросети. Файл должен удовлетворять следующим требованиям: WAV, 44100Hz, mono, 8 bit, длительностью не менее 8 секунд.

Для удобства пользователя, часть параметров можно хранить в файле конфигурации. Файл конфигурации по умолчанию называется model.yml

Переключаться между конфигурационными файлами можно с помощью опции — zoo.

Описание интерфейса пользователя

Главный экран

После запуска, программа «GunCheck» сразу готова к работе.

Искусственный интеллект в режиме реального времени обрабатывает входной видеопоток с камер наблюдения и анализирует его на предмет присутствия предметов похожих на оружие.

После определения угрозы, программа обратит внимание оператора на нужную камеру и сделает скриншот (на фото слева). Если оператор в определённый промежуток времени не предпримет никаких действий, то программа самостоятельно выполнит запрограммированное действие, например, отправит скриншот человека с оружием в мессенджер ответственному лицу, подаст сигнал тревоги на пульт охраны или заблокирует входную дверь.



На изображении главный экран программы

Слева находится область для отображения скриншотов, которые автоматически попадают в базу данных при обнаружении оружия в кадре и могут быть просмотрены позже. Если включена опция отправки, данные скриншоты, отправляются адресатам.

Основное пространство экрана занимает сетка с трансляцией видеопотока с камер наблюдения. Зелёными прямоугольниками выделяются обнаруженные программой объекты похожие на оружие. Текст наименования камеры становится красным, для привлечения внимания оператора.