Мобильное приложение непрерывной передачи видеоданных в облачные сервисы

Т. В. Лях ¹, К. В. Марченко ²

¹ *Институт автоматизации и электрометрии СО РАН*

² *Новосибирский государственный университет
Новосибирск, Россия*

Аннотация

Представлен способ получения видеодоказательств с помощью мобильного устройства – приложения для Android, позволяющего оперативно начать видеотрансляцию в облачные сервисы. Для оперативности запуска пользователю доступен виджет, который можно разместить на экране устройства. При нажатии на виджет начинается видеосъемка и отправка получаемой записи в облачный сервис, необходимая для обеспечения сохранности уже полученных данных даже в случае повреждения устройства. Для того чтобы в непредвиденной ситуации, повлекшей за собой утрату телефона, как можно больше данных оказалось сохранено на облачном сервисе, выбрана малая продолжительность одного передаваемого фрагмента – одна секунда. По завершении записи пользователь имеет возможность собрать составляющие записи воедино, при необходимости загружая недостающие фрагменты из облачного сервиса. Практическая ценность приложения заключается в том, что оно позволяет достаточно быстро и надежно фиксировать события, а собранные видеоматериалы могут быть полезны как в судебных процессах, так и для самостоятельного изучения. В статье проведен анализ существующих решений, позволяющих вести видеотрансляцию; выявлены их недостатки, затрудняющие использование этих средств для получения доказательств. Рассмотрена архитектура приложения, построенная на базе шаблона проектирования MVC и включающая в себя 6 модулей. Описана реализация приложения и показан разработанный интерфейс.

Ключевые слова

передача данных, мобильное приложение, Android, облачные сервисы, Google Диск, оперативный запуск, видеозапись

Для цитирования

Лях Т. В., Марченко К. В. Мобильное приложение непрерывной передачи видеоданных в облачные сервисы // Вестник НГУ. Серия: Информационные технологии. 2019. Т. 17, № 1. С. 53–60. DOI 10.25205/1818-7900-2019-17-1-53-60

A Mobile Application of Video Transmission to Cloud Services

T. V. Liakh ¹, K. V. Marchenko ²

¹ *Institute of Automation and Electrometry SB RAS*

² *Novosibirsk State University
Novosibirsk, Russian Federation*

Abstract

The paper presents a new way to obtain evidence with a mobile device – by means of an Android-application that allows to rapidly start video transmission to cloud services. For the sake of quick launch there is a widget available to the user which can be placed on the home screen. Tapping the widget starts video recording which is sent to cloud service as it is recorded. The latter is essential for the security of data that has already been received, even if the device has been damaged. To save as much data as possible in unexpected situations which lead to phone's loss, small duration of one fragment has been chosen – one second. Upon completion of the recording the user has an opportunity to assemble the whole clip from its fragments, downloading missing fragments from cloud service if necessary. The practical value of the application is that it allows to record events reliably and quickly enough, and the collected video

materials can be useful both for litigations and for self-study. The paper analyzes existing solutions. As a result, each of them is found to have certain disadvantages preventing them from usage in video evidence retrieval. An architecture for the application is proposed, featuring the MVC design pattern with 6 modules. The application implementation is described and the developed UI is presented.

Keywords

data transfer, mobile application, Android, cloud services, Google Drive, quick launch, video

For citation

Liakh T. V., Marchenko K. V. A Mobile Application of Video Transmission to Cloud Services. *Vestnik NSU. Series: Information Technologies*, 2019, vol. 17, no. 1, p. 53–60. (in Russ.) DOI 10.25205/1818-7900-2019-17-1-53-60

В современной российской судебной практике в качестве доказательств принимаются аудио- и видеоматериалы. Ранее процессуальное законодательство было недостаточно проработано – решение о приобщении видеоматериалов к делу в административном, уголовном, гражданском и арбитражном процессах принимал судья. Однако в апреле 2016 г. был подписан указ [1], согласно которому звуко- и видеозаписи признаются документами и выступают как доказательства в административных делах, если они имеют значение для рассмотрения и разрешения дела¹. Благодаря этому указу, рассматривая административные дела, судьи обязаны принимать к рассмотрению видеозапись, и теперь вопрос уже о признании ее достоверности.

Ведение видеозаписи полезно не только как средство фиксирования событий в потенциально опасных ситуациях, но и как превентивная мера. Если возможный правонарушитель осознает, что его действия записываются, то более вероятно, что он воздержится от правонарушения.

Видеозапись может использоваться не только для фиксации правонарушений. При соблюдении соответствующего законодательства человек вправе снимать различные эпизоды своей жизни, чтобы в дальнейшем вернуться к ним, проанализировать произошедшие события. Ему даже необязательно пересматривать записи целиком, чтобы найти что-либо интересное – уже существуют многочисленные способы автоматизированного извлечения информации из видеоматериалов. Так, например, были предложены способы обнаружения некоторых типов событий – оставление, перебрасывание предмета и пр. [2]

Доступным способом сделать видеозапись является использование мобильного телефона. Как правило, когда случается что-то экстренное, у человека под рукой нет видеокамеры. Зато есть мобильное устройство – еще в 2013 г. мобильными телефонами пользовалось 98 % российских граждан [3].

Однако при создании видеозаписи с помощью мобильного устройства существует две проблемы:

- 1) на запуск видеосъемки может затрачиваться много времени;
- 2) в случае утраты телефона не обеспечивается сохранность данных.

Таким образом, актуальными задачами являются выбор способа быстрого запуска видеозаписи с мобильного устройства и создание средств, обеспечивающих сохранность видеозаписи даже при разрушении мобильного устройства.

В статье описано мобильное приложение, которое обеспечивает оперативность запуска записи с помощью виджета на экране и сохранность видеозаписи за счет непрерывной передачи видеофрагментов длительностью 1 секунда в облачные сервисы.

Авторами статьи был выбран определенный набор используемых технологий. Для обеспечения сохранности видеозаписи было предложено отправлять запись на удаленные серверы непосредственно в процессе ее создания. Удаленные серверы должны удовлетворять двум основным требованиям – они должны быть надежными и предоставлять минимум 5 Гб хра-

¹ О внесении изменения в ст. 26.7 Кодекса Российской Федерации об административных правонарушениях в части обязательности отнесения материалов фото- и киносъемки, звуко- и видеозаписи к доказательствам по делу об административном правонарушении: ФЗ от 26.04.2016 № 114-ФЗ // Рос. газета. 2016. 27 апр. С. 21.

нилища без взимания платы. Таким требованиям удовлетворяют многие облачные сервисы, вследствие чего они и были выбраны как место, куда транслируются данные пользователя.

Одним из возможных решений передачи видеозаписи на удаленные серверы во время ее создания является потоковое вещание (англ. *streaming*). К примеру, можно было бы использовать набор технологий WebRTC, который позволил бы работать с видеопотоками через браузер [4]. Однако многие облачные сервисы не поддерживают потоковое вещание, поэтому был выбран иной подход. Каждую секунду очередной созданный фрагмент видеозаписи сохраняется локально, после чего отправляется в облачный сервис. Таким образом, сохранность видеоданных достигается за счет отправки большого количества видеофрагментов малой длины, которые впоследствии могут быть собраны в единую запись.

Среди множества облачных сервисов для хранения данных выбор пал на Google Диск, поскольку он гарантирует сохранность данных и предоставляет 15 Гб для хранения данных бесплатно.

Мобильной платформой, под которую разработано приложение, выбран Android. По статистике на первый квартал 2017 г., 85 % мирового рынка мобильных устройств занимают Android-устройства², поэтому ориентация на данный сегмент представляется достаточной.

Главное требование, предъявленное к способам запуска трансляции, – запись должна начинаться как можно быстрее. В связи с этим сделан выбор в пользу использования виджета. Пользователь может расположить виджет на любом экране мобильного устройства, в том числе и на главном, и тогда ему не нужно тратить время на поиск приложения в общем списке и нажатие кнопки в запущенном приложении – достаточно один раз нажать на виджет.

В настоящее время существуют мобильные приложения под различные платформы, занимающиеся передачей видеоданных на удаленные серверы. Был произведен анализ известных Android-приложений, наиболее приближенных к решению упомянутых проблем с получением и сохранностью видеоматериалов (рис. 1).

	Непрерывная передача	Локальное сохранение при отсутствии Internet	Продолжение записи при блокировке	Оперативный запуск	Сохранность видеозаписи длительностью менее 5 секунд	Возможность получить единую запись
YouTube	+	–	–	–	+	?
Instagram	+	–	–	–	+	+
Periscope	+	–	–	–	+	+
ОКО	+	+	+	+	–	–

Рис. 1. Сравнение существующих решений

Fig. 1. Comparison of existing solutions

² Smartphone OS. URL: <https://www.idc.com/promo/smartphone-market-share/os>.

Такие популярные сервисы, как YouTube, Periscope, Instagram, предоставляют возможность делать потоковую видеозапись так, что ее могут увидеть пользователи других устройств. Соответствующие приложения хорошо знакомы широкой публике и обладают высокой степенью надежности. Однако все они были созданы для иных целей, вследствие чего обладают рядом особенностей, которые являются недостатками в контексте цели разработанного приложения. Это, во-первых, отсутствие локального сохранения при невозможности получить доступ в Интернет и, во-вторых, достаточно долгий запуск: чтобы запустить трансляцию, пользователю как минимум нужно запустить приложение и нажать в нем соответствующую кнопку.

Этих недостатков лишено ОКО³, которое позволяет начать запись при нажатии на виджет. Однако и у него есть свои недостатки.

1. Отсылка фрагментов видеозаписи происходит каждые 5 или 10 секунд, что является достаточно большим промежутком времени. Если устройство будет повреждено на 4-й или 9-й секунде, есть шанс потерять важные данные.

2. Отсутствие возможности получить единую запись из снятых фрагментов. Приложение способно проиграть фрагменты один за другим, однако функция создания цельной видеозаписи отсутствует.

Главная функция разработанного приложения – создание видеозаписи и передача ее в облачный сервис. При этом запись передается по фрагментам: каждую секунду очередной фрагмент видеозаписи сохраняется локально и отправляется в облачный сервис (рис. 2).

Рис. 2. Блок-схема
общего алгоритма работы приложения
Fig. 2. Flowchart of application's general algorithm

³ ОКО. URL: <http://oko.oprg>.

После запуска приложения и авторизации пользователю доступен предварительный просмотр камеры и кнопка «Начать запись». При нажатии на кнопку начинается видеозапись и ее передача по фрагментам в отдельном, не главном, потоке. Этот процесс продолжается до тех пор, пока пользователь не нажмет на кнопку «Стоп».

Если пользователь уже авторизован, для начала записи ему достаточно нажать на виджет, не заходя предварительно в приложение.

Если в процессе ведения записи пользователь временно потерял доступ в Интернет, после повторного подключения записанные во время автономного режима работы фрагменты все равно будут отправлены в Google Диск, что обеспечивается сервисами Google. Так же, к примеру, функционирует и Google Analytics [5].

Разработанная архитектура приложения включает в себя 6 модулей (рис. 3).

1. Модуль авторизации выполняет необходимую для работы с Google Диском авторизацию пользователя через аккаунт Google.

2. Модуль ведения видеозаписи занимается получением видеоданных с камеры, разбиением их на фрагменты и локальным сохранением этих фрагментов.

3. Модуль трансляции фрагментов отправляет во время ведения записи уже сохраненные локально фрагменты в облачный сервис.

4. Модуль управления записями служит для предоставления пользователю возможности просматривать имеющиеся видеозаписи, сведения о дате их создания, их местоположение (локально и/или в облачном сервисе) и состояние (фрагментирована запись или едина). Также дает возможность удалить запись как локально, так и с облачного сервиса.

5. Модуль сборки записи собирает фрагменты в единую видеозапись.

6. Модуль синхронизации записей с облачным сервисом позволяет загрузить недостающие фрагменты с облачного сервиса на мобильное устройство.

Рис. 3. Архитектура ПО

Fig. 3. Software architecture

Рассмотрим особенности реализации приложения. Минимальная поддерживаемая версия операционной системы – Android 5.0 Lollipop, что позволяет охватить 79,8 % от всех Android-устройств (по состоянию на декабрь 2017 г.)⁴. Дизайн учитывает Material Design⁵, разработанный компанией Google (рис. 4). Создаваемая видеозапись имеет формат MP4, ее кадровая частота (FPS) равна 30 кадров в секунду. Время между ключевыми кадрами составляет одну секунду.

Рис. 4. Экран ведения записи и экран списка всех записей

Fig. 4. Recording and all records screens

Приложение реализовано на языке Java. Архитектура разработана на базе шаблона MVP (Model-View-Presenter), что делает ее гибкой и позволяет добавить новые способы запуска видеотрансляции без модификации существующего кода.

Для работы с камерой используется android.hardware.camera2 API, пришедший на смену Camera API с уровня Android API 21.

Для получения с камеры видеопотока, отображения его на дисплее устройства и одновременно с этим разбиения видеопотока на фрагменты за основу была взята библиотека AudioVideoRecordingSample⁶, переработанная в собственный переиспользуемый компонент. Для рендеринга видеопотока применяется OpenGL ES.

⁴ Distribution dashboard. URL: <https://developer.android.com/about/dashboards/index.html>.

⁵ Material Design для Android. URL: <https://developer.android.com/design/material/?hl=ru>.

⁶ AudioVideoRecordingSample. URL: <https://github.com/saki4510t/AudioVideoRecordingSample>.

Чтобы создавать видеофайлы-фрагменты, была предпринята попытка использовать MediaRecorder API, однако выявилась проблема. Управление записью в классе MediaRecorder основано на простом конечном автомате. После окончания записи одного фрагмента вновь перейти в состояние записи можно лишь снова проведя инициализацию, конфигурацию источника данных и подготовку (каждому из этих действий соответствует отдельное состояние). Все эти операции занимают время порядка 700 миллисекунд [6], а поскольку, согласно требованиям, перезапуск должен осуществляться каждую секунду, многие моменты оказываются незафиксированными. Поэтому для видеозаписи используются более низкоуровневые классы MediaCodec и MediaMuxer. MediaCodec используется для сжатия видеопотока, в то время как MediaMuxer записывает видеопоток в MP4-файл.

Для того чтобы видеофрагменты отправлялись в облачные сервисы в отдельных потоках, применяется RxJava 2, в частности планировщики Schedulers. Работа с Google Диском осуществляется через Google Drive Android API. С целью создать на основе снятых видеофрагментов цельную видеозапись применяется написанная на языке программирования Java библиотека mp4parser⁷, созданная специально для работы с MP4-файлами.

Резюмируем: представленное Android-приложение является удобным способом сбора видеoinформации, которая может не только использоваться для личных нужд, но и выступать в качестве доказательства в судебных разбирательствах. В целях обеспечения сохранности данных в процессе создания видеозаписи ведется ежесекундное локальное сохранение полученного фрагмента и его передача в облачный сервис Google Диск, где любой зарегистрированный пользователь может получить пространство достаточного размера для хранения данных бесплатно. Для того чтобы не пропустить важные моменты, приложение снабжено способом оперативного запуска видеотрансляции с помощью виджета, который пользователь может разместить на экране там, где ему удобно.

Приложение выгодно выделяется среди конкурентов небольшой продолжительностью создаваемого и отправляемого видеофрагмента, что при внезапной поломке устройства в процессе видеосъемки позволяет сохранить больший объем данных. Кроме того, поскольку работать с множеством фрагментов вместо единой записи по меньшей мере неудобно, предусмотрена возможность произвести сборку фрагментов в один MP4-файл. При этом, если на мобильном устройстве не хватает каких-либо фрагментов, имеющихся в облачном сервисе, они загружаются на устройство перед сборкой.

В настоящий момент ввиду малой продолжительности самостоятельных фрагментов собираемая запись имеет достаточно большой размер. В дальнейшем планируется работа по его уменьшению. Помимо этого, приложение передает данные только в облачный сервис Google Диск. При поддержке иных облачных сервисов (таких как Яндекс.Диск, Dropbox) была бы удобна функция автоматического переключения на другой облачный сервис при достижении лимита в каком-то хранилище.

В перспективе возможно добавление запуска видеозаписи по голосовой команде, а также введение системы оповещений в виде push-нотификаций при начале записи кругу доверенных лиц с предоставлением им доступа к создаваемой видеозаписи. Push-нотификации могут показаться недостаточно привлекающим внимание способом сообщить о важном событии, однако их использование в этом качестве активно применяется на практике [7].

Список литературы / References

1. **Тимошенко В.** Кинопоказ в суде: когда видеозапись является доказательством // Гарант: справ.-правовая система. 2016. URL: <http://www.garant.ru/article/733154> (дата обращения 21.05.2018).
Timoshenko V. Showing video in court: when videos are evidence. URL: <http://www.garant.ru/article/733154> (accessed 21.05.2018). (in Russ.)

⁷ Java MP4 Parser. URL: <https://github.com/sannies/mp4parser>.

2. **Гордин М. С., Иванов С. А.** Алгоритмы обнаружения тревожных событий для систем автоматизированного видеонаблюдения // Вестник НГУ Серия: Информационные технологии. 2017. Т. 15, № 3. С. 21–30. DOI 10.25205/1818-7900-2017-15-3-21-30
Gordin M. S., Ivanov S. A. Algorithms for Detecting Incident for Automated Video Surveillance Systems. *Vestnik NSU. Series: Information Technologies*, 2017, vol. 15, no. 3, p. 21–30. (in Russ.) DOI 10.25205/1818-7900-2017-15-3-21-30
3. **Морозова А.** Сайт в облаке // Российская Бизнес-газета – Инновации. 2013. № 917 (39). URL: <https://rg.ru/2013/10/08/telefoni.html> (дата обращения 21.05.2018)
Morozova A. A website in the cloud. *Rossiyskaya Biznes-Gazeta – Innovatsii*, 2013, no. 917 (39). URL: <https://rg.ru/2013/10/08/telefoni.html> (accessed 21.05.2018). (in Russ.)
4. **Архипец И. А., Бежецов Д. Е., Данилова Ю. Э., Кандров Д. Ю., Солодовник М. Н., Федотов А. А.** Открытые решения веб- / видеоконференцсвязи и проект OpenMeetings // Вестник НГУ Серия: Информационные технологии. 2018. Т. 16, № 1. С. 24–38. DOI 10.25205/1818-7900-2018-16-1-24-38
Arkhipets I. A., Bezhetskov D. E., Danilova Yu. E., Kandrov D. Yu., Solodovnik M. N., Fedotov A. A. Open Source Web- / Videoconferencing Solutions and OpenMeetings Project. *Vestnik NSU. Series: Information Technologies*, 2018, vol. 16, no. 1, p. 24–38. (in Russ.) DOI 10.25205/1818-7900-2018-16-1-24-38
5. **Демиш В. О., Пищик Б. Н.** Автономная работа Android-приложений и алгоритмы синхронизации данных // Вестник НГУ Серия: Информационные технологии. 2014. Т. 12, № 3. С. 25–33.
Demish V. O., Pischik B. N. Offline Android Application and Algorithms of Data Synchronization. *Vestnik NSU. Series: Information Technologies*, 2014, vol. 12, no. 3, p. 25–33. (in Russ.)
6. **Довбня Д.** Запись и обработка видео на Android // Хабр. 2015. 9 дек. URL: <https://habr.com/post/272705> (дата обращения 21.05.2018)
Dovbnya D. Recording and processing video on Android. *Habr*, 2015, December 9. URL: <https://habr.com/post/272705> (accessed 21.05.2018). (in Russ.)
7. **Букшев И. Е.** Medilux – сервис интеллектуального формирования расписания посещений медицинских учреждений // Вестник НГУ Серия: Информационные технологии. 2018. Т. 16, № 2. С. 41–48. DOI 10.25205/1818-7900-2018-16-2-41-48
Bukshev I. E. Medilux – Service of Intellectual Forming the Schedule of Visiting Medical Institutions. *Vestnik NSU. Series: Information Technologies*, 2018, vol. 16, no. 2, p. 41–48. (in Russ.) DOI 10.25205/1818-7900-2018-16-2-41-48

Материал поступил в редколлегию

Received
20.07.2018

Сведения об авторах / Information about the Authors

Лях Татьяна Викторовна, инженер-программист, Институт автоматики и электрометрии СО РАН (пр. Академика Коптюга, 1, Новосибирск, 630090, Россия)

Tatiana V. Liakh, Software Engineer, Institute of Automation and Electrometry SB RAS (1 Academician Koptyug Ave., Novosibirsk, 630090, Russian Federation)

antsys_nsu@mail.ru

Марченко Кристина Витальевна, магистрант факультета информационных технологий Новосибирского государственного университета (ул. Пирогова, 1, Новосибирск, 630090, Россия)

Kristina V. Marchenko, Master's Student, Faculty of Information Technologies, Novosibirsk State University (1 Pirogov Str., Novosibirsk, 630090, Russian Federation)

k.marchenko@g.nsu.ru