

# Точная синхронизация записи видео на Android-смартфонах

Ахметьянов Азат, группа 18.Б11-мм  
12 июня

Научный руководитель: ст. преп. Я.А. Кириленко  
Консультант: Инженер-исследователь, Сколтех А.В. Корнилова

СПбГУ, 2021

# Смартфон как сенсор

Современные смартфоны содержат большое количество сенсоров, что позволяет в некоторых случаях использовать их вместо специализированного оборудования.

- Мобильные сенсоры используются для анализа активности и мониторинга здоровья
- Для распределенной системы мониторинга вибраций в здании [1]
- **Использование данных камер (видео, фото) для 3D реконструкции динамических сцен, SLAM [2]**

[1] [Design and Validation of Android Smartphone Based Wireless Structural Vibration Monitoring System, 2020](#)

[2] [Stereo vision based indoor/outdoor navigation for flying robots, 2013](#)

# Синхронизированное видео

**Синхронизированное видео с нескольких камер** используется в:

- SLAM
- Склейка панорамного видео
- Deblurring
- ...

Системы стереозрения — дорогие, хорошо было бы и тут применять смартфоны

# Синхронизация времени

Существует множество методов синхронизации:

- **Аппаратная синхронизация**

- Аппаратные триггеры

- **Программная синхронизация**

- **Сетевые протоколы**

- NTP/PTP/RBS

- используют обмен сообщений в сети, их точность зависит от качества сети и задержек передачи сообщений

- **Синхронизация, основанная на данных**

- По видео с камер [5]

- Основанная на звуке [6]

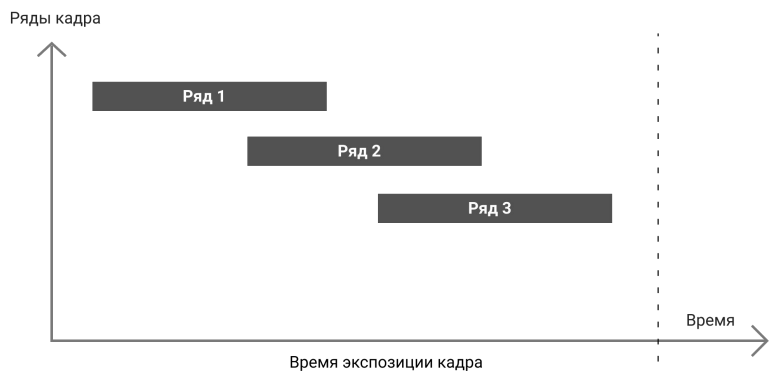
- По показаниям сенсоров (IMU)

[5] [Rolling Shutter Camera Synchronization with Sub-millisecond Accuracy, 2019](#)

[6] [Self-localization and channel synchronization of smartphone arrays using sound emissions, 2016](#)

# Особенности работы с Android-устройствами

- Доступная функциональность управления камерами очень различается в зависимости от уровней API/версий Android
- API камеры выдает наносекундные временные метки для кадров, но при этом **запросить съемку в точный момент времени нельзя**
- Сенсоры мобильных камер считывают изображение не единым кадром, а по рядам - эта технология называется rolling shutter [7]



[7] [Rolling shutter, Wikipedia](#)

# Libsoftwaresync [Wireless Software Synchronization of Multiple Distributed Cameras, 2019](#)

## Проблемы

- Поддерживается **только фотосъемка**
- Запускалось только на Pixel смартфонах, не очень приспособлено к массовому использованию
- Конфигурационные параметры под устройство **нужно прописывать вручную** (например, период потока кадров)

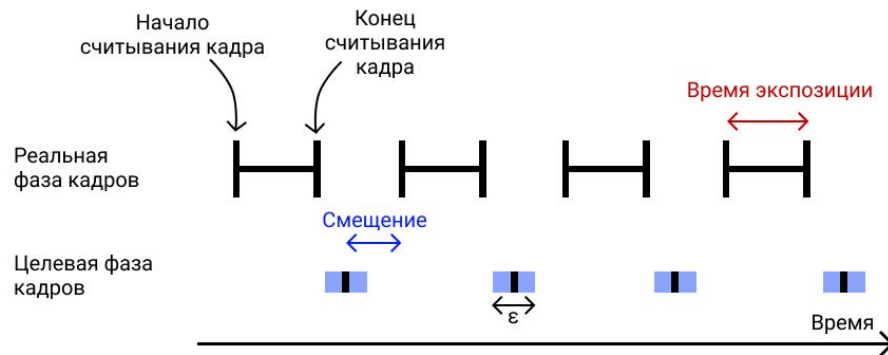


Рис. Иллюстрация проблемы смещения фазы кадров. Приложение от Google Research решает ее при помощи вставки кадров с отличающимся временем экспозиции.

# Цели и задачи

**Целью** работы является создание Android-приложения для записи синхронизированных видео с нескольких смартфонов и дальнейшего использования в перечисленных областях применения.

## **Задачи:**

- Разработка алгоритма для оценки периода временных меток кадров
- Анализ стабильности фазы временных меток кадров при переключении режимов камеры (*превью* → *видео*)
- Реализация записи синхронизированного видео в приложении на Android
- Оценка качества синхронизации видео с использованием разработанного приложения
- Апробация на склейке панорамного видео

# Анализ стабильности фазы кадров

Нужно, чтобы понять, **влияет ли старт видео на стабильность фазы кадров** (если да, то может произойти рассинхронизация)

Реализовано приложение, сохраняющее временных меток потока кадров до/после начала записи видео

- Запуск на устройствах из firebase (сервис для тестирования от Google) + скрипт для выгрузки данных
- [возможно другие сервисы]
- Условия записи:
  - период превью длительностью 15 секунд
  - запись видео длительностью 1 минута



# Эксперименты в Firebase

Общее количество моделей в Firebase: 180+

Требования для нашего эксперимента:

- Camera 2 API
- Level support >= Limited

Было отобрано подмножество устройств, соответствующих требованиям



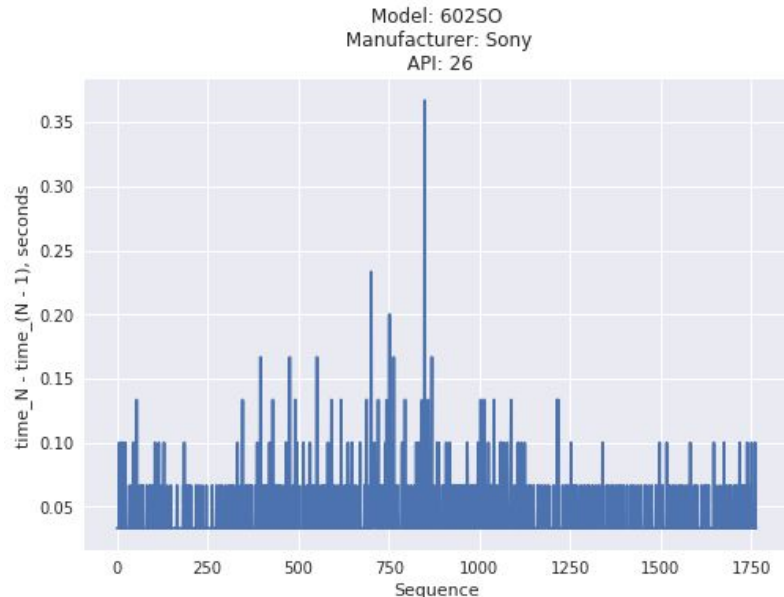
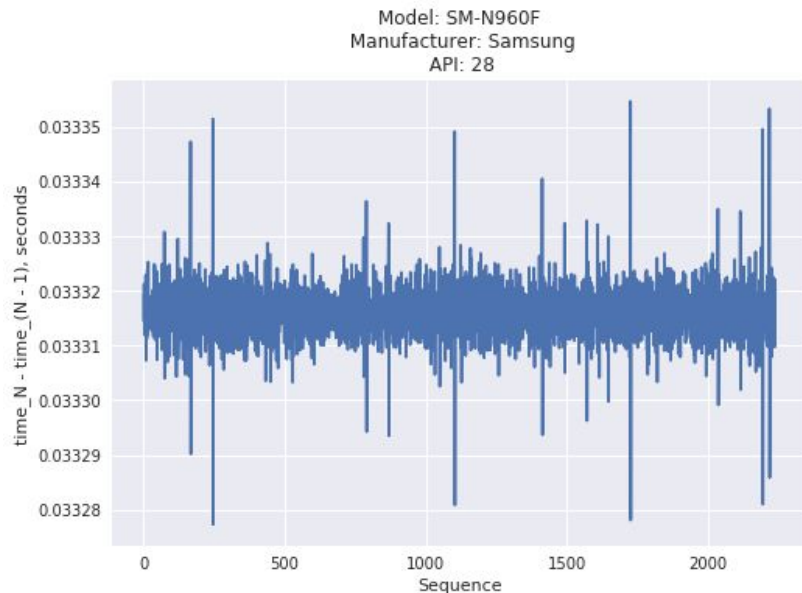
# firebase

Производитель	Кол-во
Google	5
LG	7
Samsung	20
Sony	4
Xiaomi	3
Прочие	13
<b>Всего</b>	<b>52</b>

Статистика моделей,  
удовлетворяющих требованиям

# Временные метки кадров в данных из Firebase

- На графиках разность  $N$ -й и  $(N-1)$ -й временной метки кадров
- **Выбросы как на правом графике на фазу не влияют**



# Алгоритм оценки точного периода кадров

После анализа данных с устройств в figebase, была предложена следующая математическая модель фазы временных меток ( $\epsilon$  - погрешность,  $T$  - период временных меток):

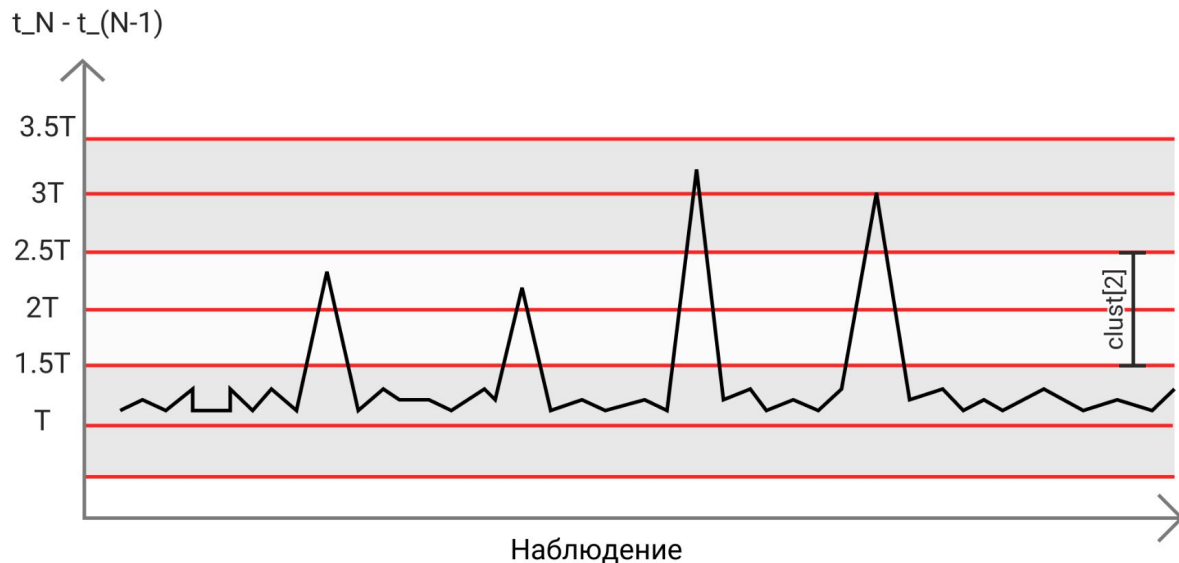
$$t[i] = t[0] + N(i)T + \epsilon$$

$$N(i) \in \mathbb{N}$$

$$N(i) < N(i + 1)$$

# Алгоритм оценки периода кадров

Для оценки периода был предложен метод weighted clust, основной принцип работы изображен на рисунке:



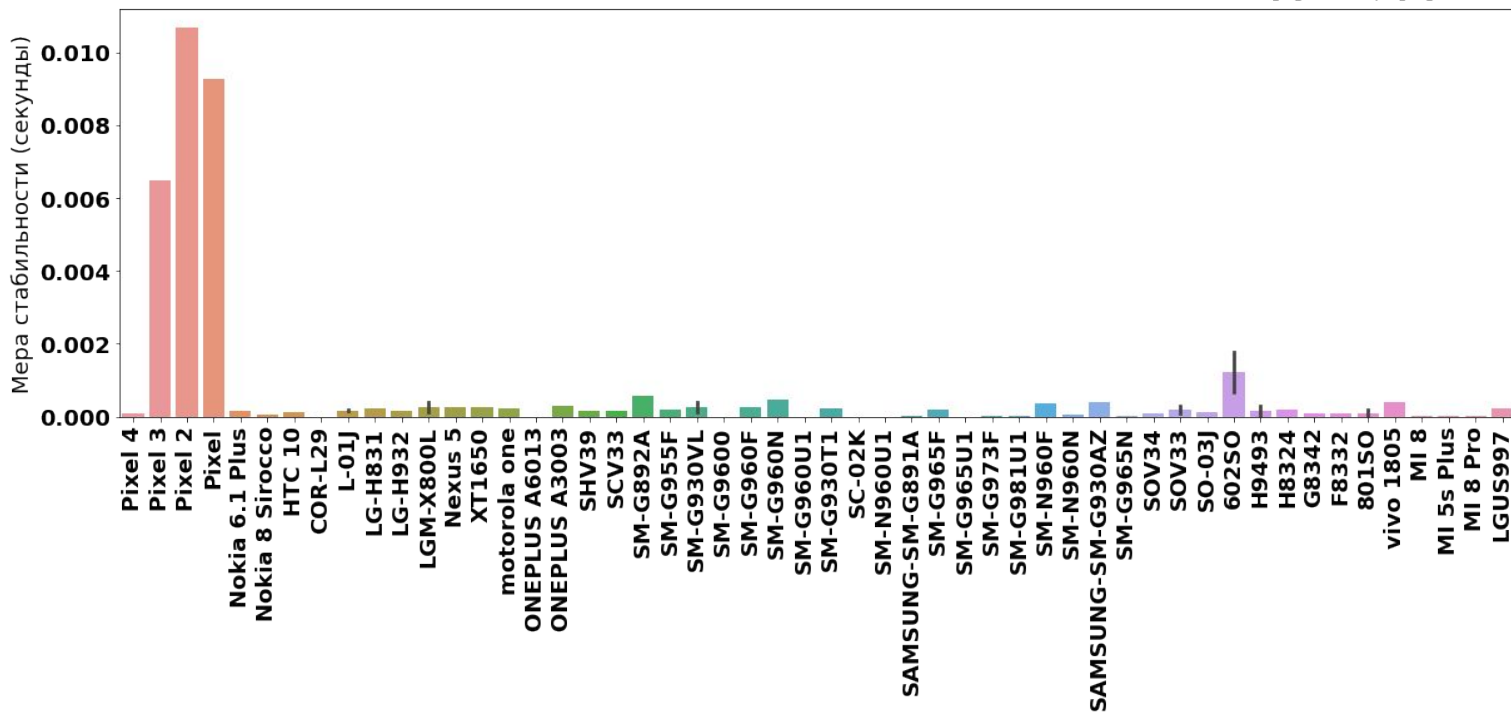
$$T_{init} = \min(\Delta t[i])$$
$$T = \left( \sum \frac{\text{median}(clust_i) |clust_i|}{i} \right) / N$$

Получаем оценку периода  $T$

# Анализ стабильности на данных из Firebase

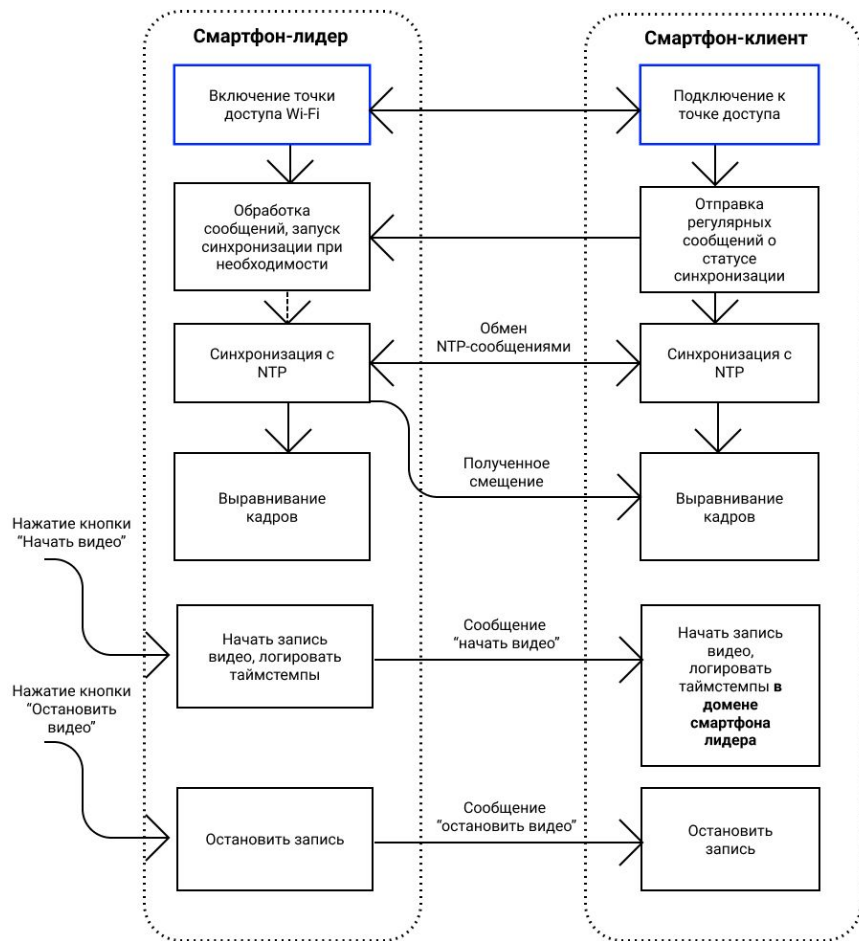
Стабильность оценивалась по формуле:  $std(t[i] - (t[0] + N(i) * T))$

где  $T$  - оценка периода,  $t$  - временные метки кадров, а  $N(i)$  такие, что  $0 < t[i] - (t[0] + N(i)T) < T$



# Пайплайн решения с модифицированным libsoftwaresync

- Новый тип сообщений “начать видео” и “остановить видео”
- Логирование временных меток кадров при записи
- Дополнительно:
  - Скрипты для разделения пар полученных видео с csv-логами их временных меток **на кадры и их сопоставления**
  - Автоматизация подсчета **конфигурационных параметров** в приложении, таких как период кадров, под конкретное устройство



# Запуск приложения с видео

Модифицированное приложение использовалось на двух Samsung Galaxy S20 с таким стендом

Samsung Galaxy S20

- Camera 2 API
- Стандартная задняя камера
- Разрешение камеры: 12.0 МП

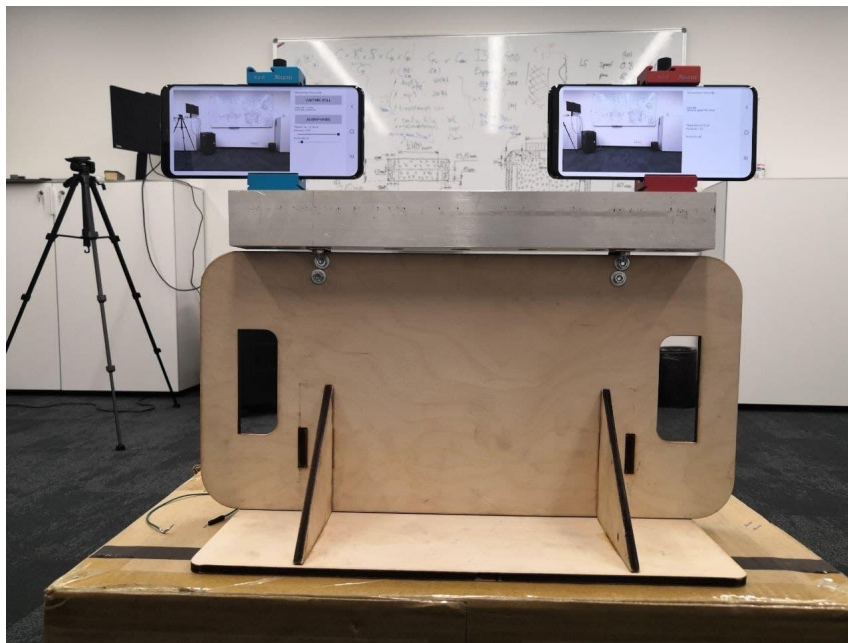


Рис. Стенд для экспериментов с синхронизированной записью

# Методология оценки качества синхронизации

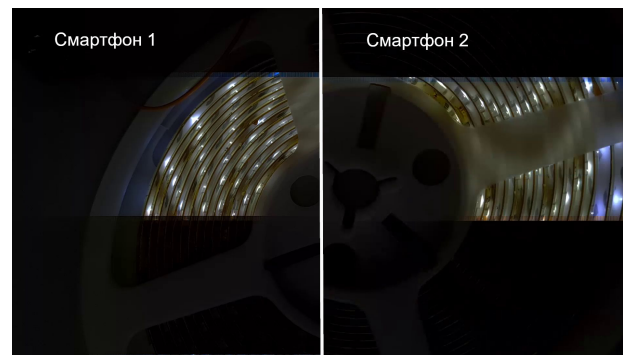
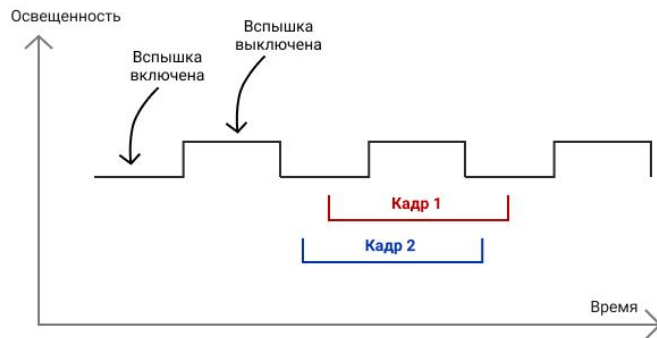


Рис. Пример пары кадров из записанного приложением видео



# Результаты эксперимента со вспышкой

Samsung Galaxy S20, exposure time: 1/16000. Время считывания одного ряда ~ 12 микросекунд  
Данные по 5 парам видео длительности ~1 минута, CI=0.95

Номер эксперимента	Количество пар кадров	Ошибка синхронизации, мксек
1	1098	$67.07 \pm 36.25$
2	1183	$58.55 \pm 0.00$
3	1167	$116.99 \pm 2.29$
4	1156	$58.55 \pm 0.00$
5	1054	$144.92 \pm 87.20$

# Апробация на склейке панорамного видео

- Реализован скрипт с использованием инструмента Hugin Panorama Tools + ffmpeg
- Записано тестовое видео с быстрым движением

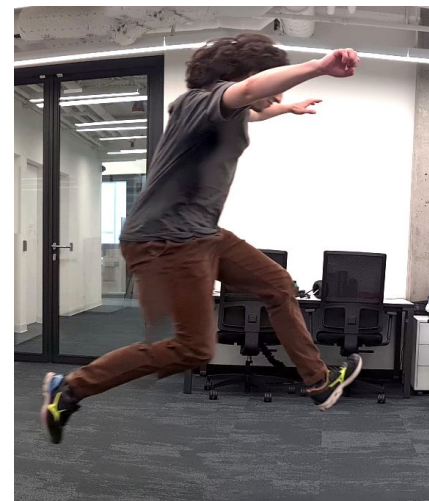
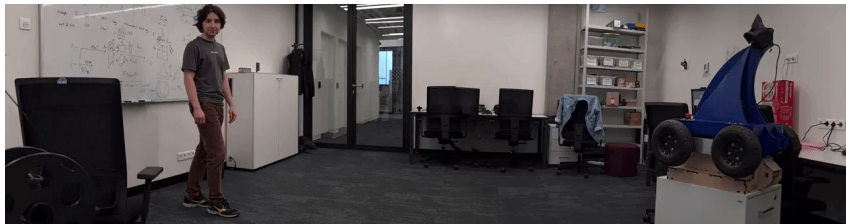


Рис. На левом изображении часть панорамы с синхронизированными кадрами, на правом - панорама из кадров со смещением в 33 миллисекунды

# Результаты

- Предложен алгоритм для оценки периода кадров
- Проведен **анализ стабильности фазы временных меток** кадров на 52 устройствах из figebase, полученные результаты показывают смещение фазы на **не более миллисекунды за 1 минуту** на всех устройствах, кроме нескольких Pixel смартфонов
- Реализована запись **синхронизированных видео** в приложении на Android
- Проведена оценка качества синхронизации видео с использованием разрабатываемого приложения
- Выполнена апробация на склейке панорамы