|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | http://mai.ru/life/brand/mai.gif | 1. Федеральное государственное бюджетное образовательное учреждение 2. высшего образования 3. "МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ 4. (НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)" | |
| Институт №3 «Системы управления, информатика и электроэнергетика» |
| Кафедра 308 «Информационные технологии» |

|  |
| --- |
| **ОТЧЕТ**  **По лабораторной работе** |
|  |
| «Распознавание списка присутствующих» |
| |  |  |  | | --- | --- | --- | | Студент группы  М30-321Б-17 | *« »*  2019 г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *(Игнатьев Андрей Александрович)* | |  |  |  | | Принял | «\_\_»\_\_\_\_\_\_\_2019 г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *(Максимов А.Н.)* | |

Москва**.**2019

**СОДЕРЖАНИЕ.**

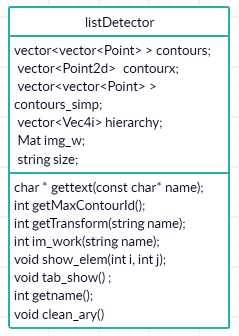
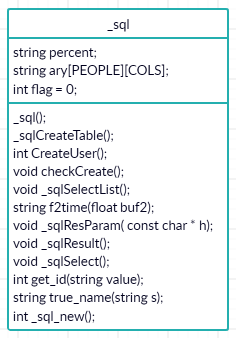
1. **Введение……………………………………………………..……………………………...3**
2. **Ход работы………………..…………………………………………………………………4**
3. **Класс \_sql ……………….……………………………………..……………………………6**
4. **Класс listDetector………………….…………………………………………………….7**
5. **Скриншоты работы программы………….…………………………………….8**
6. **Результаты работы программы………………………………………………...10**
7. **Заключение…………………………………………..…………..………………………15**
8. **Список Источников…………………………………………………………...........16**

**ВВЕДЕНИЕ.**

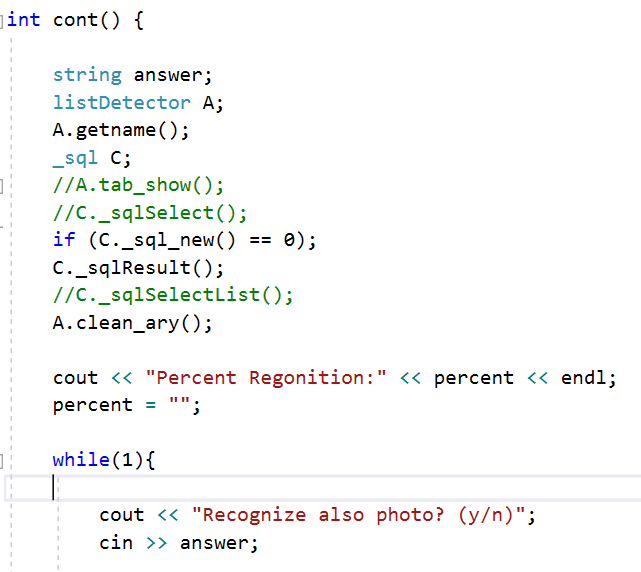
Была поставлена задача распознать отсканированный список. Выявить пришедших, опоздавших и не явившихся людей. Для распознавания текста использовались библиотеки OpenCV, TesseractOCR и SQLite3.Количество человек в списке = 13. Количество столбцов = 4.

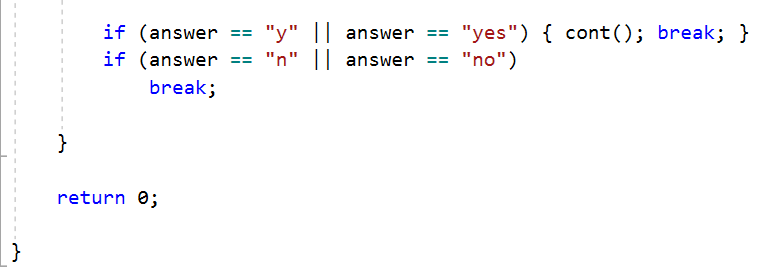
**1. Ход работы.**

В ходе работы было создано 2 класса: \_sql(отвечающий за работу с базой данных) и listDetector(отвечающий за распознавание текста на изображении);

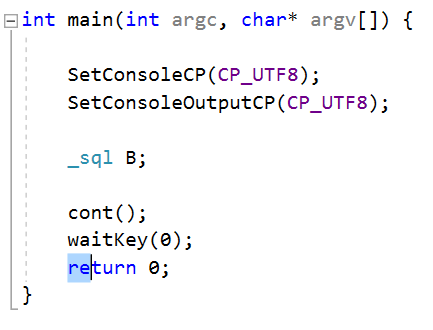


Работа с классами происходит в функции cont(), после окончания распознавания изображения, пользователь получает возможность распознать еще одно. (рекурсивный вызов cont()).





В функции main происходит вызов функции cont().



**2. Класс \_sql.**

Конструктор класса \_**sql()** создает подключение к базе данных.

Функция **\_sqlCreateTable()** создает 2 таблицы (если они еще не созданы):

-таблица Users, содержащая в себе ФИО участников конференции. (2 поля int id PRIMARY KEY, VARCHAR ini)

-таблица ListUs в которую заносятся результаты распознавания изображения.(4 поля int id PRIMARY KEY, int nameid, double time, int stat). В поле статус хранится информация о том, пришел ли человек.

Функция **CreateUser()** заполняет таблицу Users 13-ю пользователями.

Функция **checkCreate()** проверяет необходимость работы функций \_sqlCreateTable() и CreateUser().

Функция **f2time**() представляет время в читаемом виде, переводя float в string и работая со строкой.

Функция **\_sqlResParam()** получает данные из бд по параметру и выводит их на экран (параметры определены: пришел вовремя, опоздал, не пришел)

Функция **\_sqlResult()** вызывает функцию \_ sqlResParam() со всеми 3-мя параметрами.

Функция **getid()** возвращает id пользователя из таблицы Users по параметру ini.

Функция **get\_true\_name()** возвращает строку с ФИО без помех.

Функция **\_sql\_new()** заносит результаты распознавания текста в таблицу ListUs

**3. Класс listDetector**.

Конструктор класса listDetector() вызывает функцию getname()

Функция getname() запрашивает у пользователя имя изображения и проверяет изображение на существование.

Функция gettext() используя библиотеку TesseractOCR находит весь текст на изображении.

Функция getMaxContourId() возвращает индекс самого длинного контура из массива контуров.

Функция getTransform() производит гомографию исходноко изображения и получает выровненное по таблице изображение.

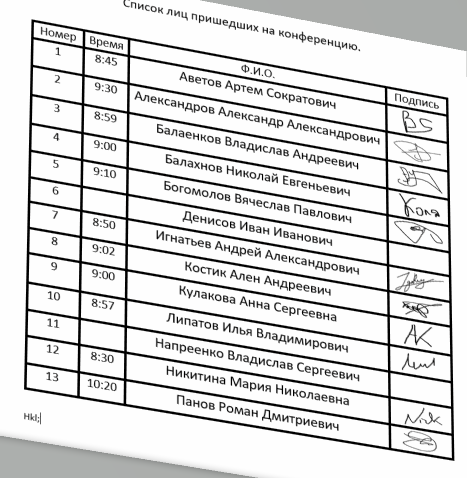
Функция imwork() преобразует полученное в результате работы getTransform изображение и разбивает его на контуры в которых ищется текст.

Функция showElement() выводит на экран элемент массива распознанного текста.

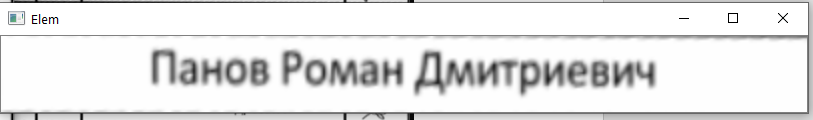
Функция tabShow() выводит в консоль все распознанные записи.

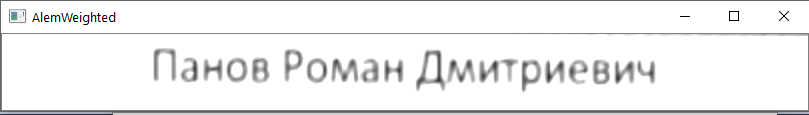
Функция clean\_ary() очищает массив распознанного текста.

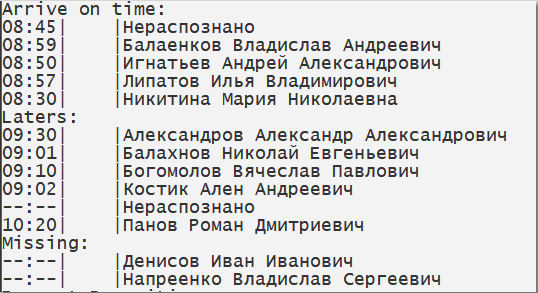
**4. Скриншоты работы программы.**

****







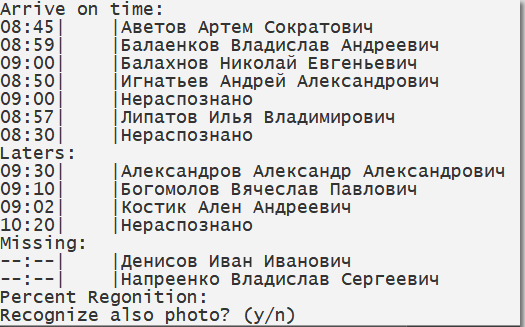


**5. Результаты работы программы.**

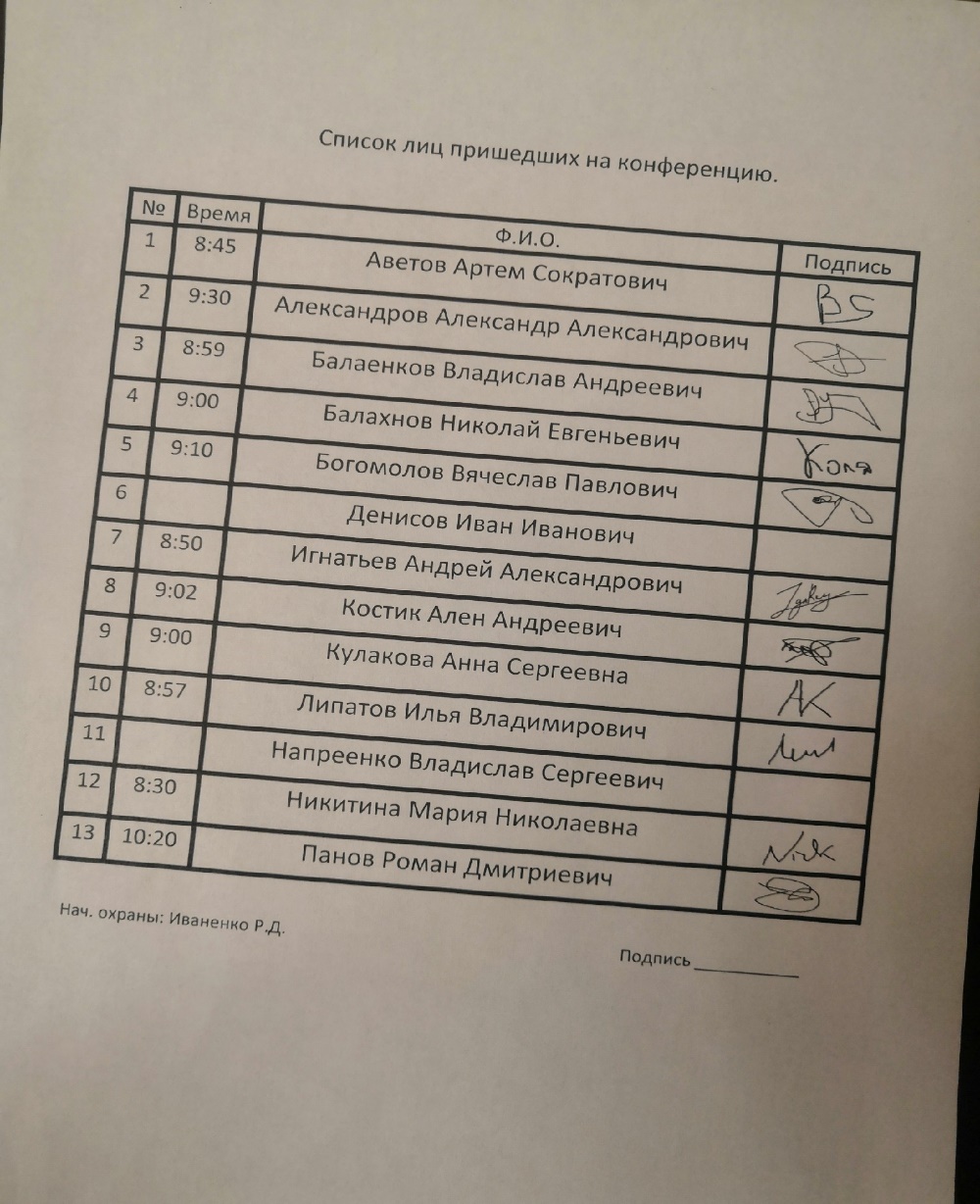
**Вход программы (1.PNG):**

****

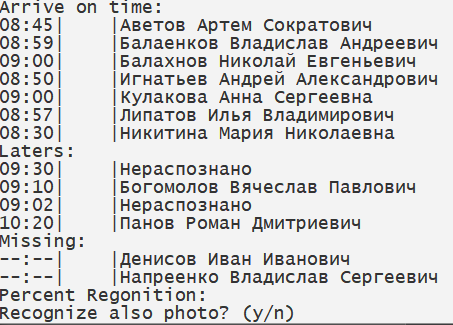
**Выход программы:**



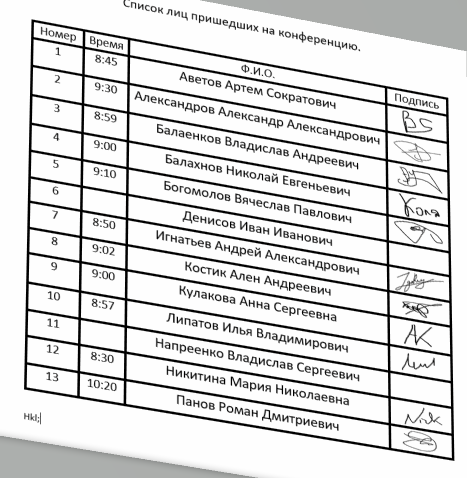
**Вход программы(1.jpg):**

****

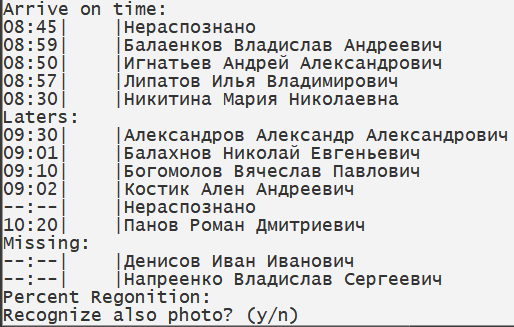
**Выход программы:**



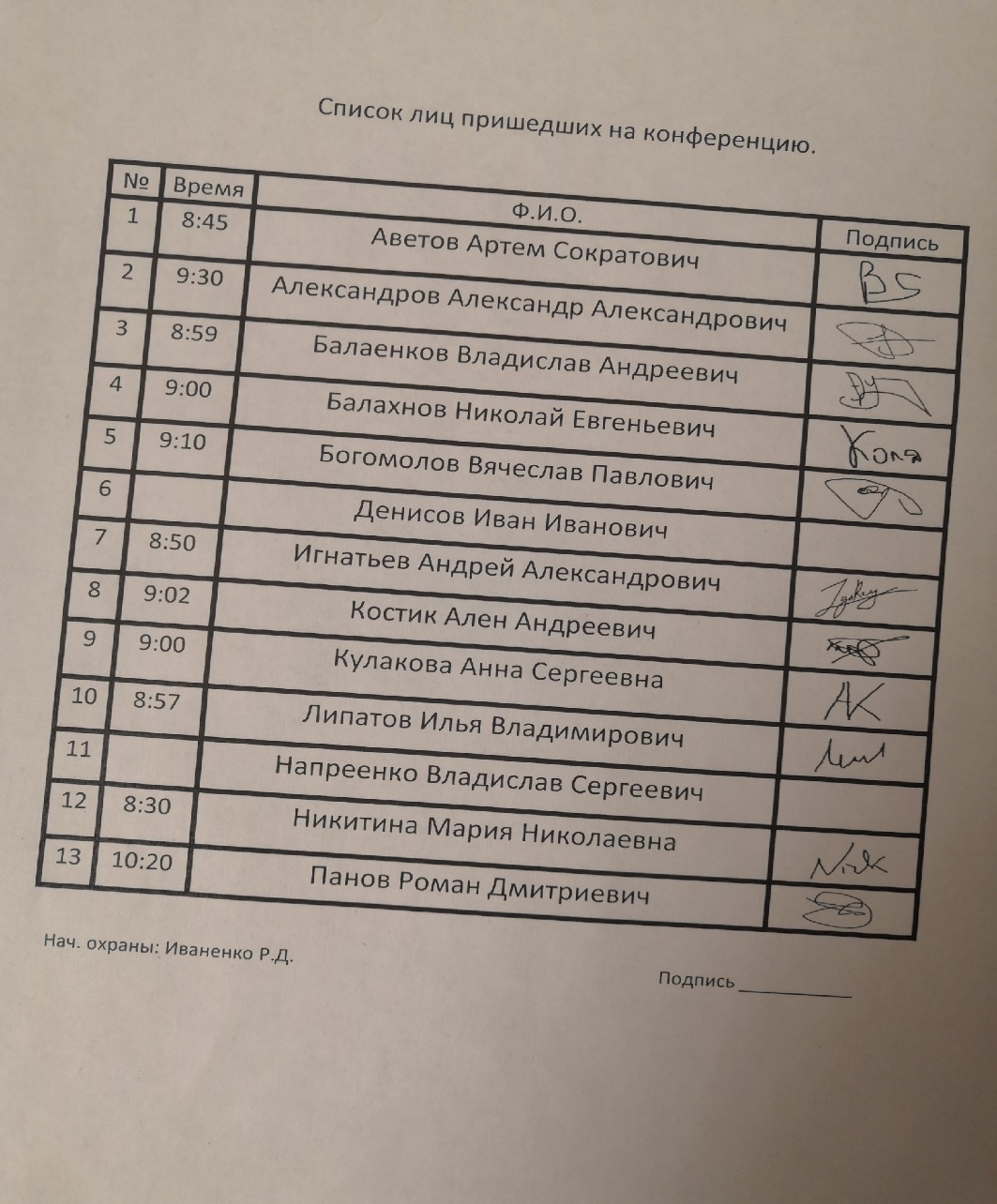
**Вход программы (2.PNG)**

****

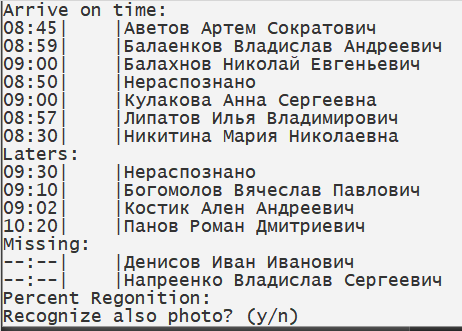
**Выход программы:**



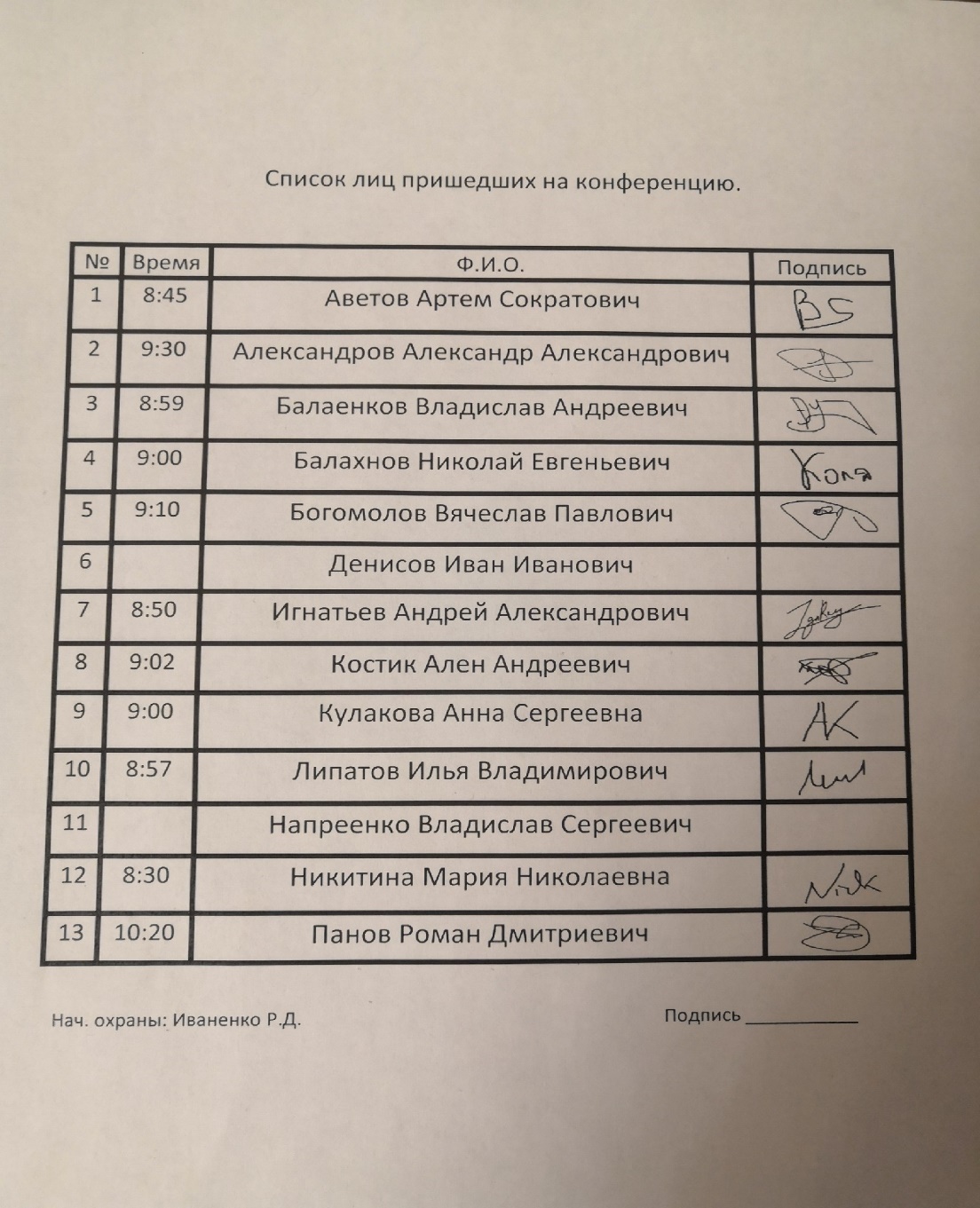
**Вход программы (3.jpg):**

****

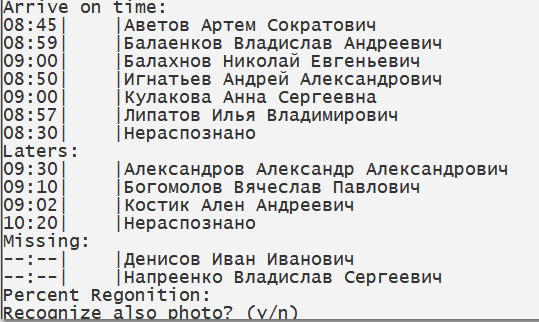
**Выход программы:**



**Вход программы(4.jpg)**

****

**Выход программы:**



**ЗАКЛЮЧЕНИЕ**

В результате работы было достигнуто идеальное распознавание отсканированных списков. Чуть меньшее качество распознавания было получено для фотографий.

Были получены навыки работы с библиотеками openCV, TesseractOCR и SQLite3.

**Список источников**

<https://github.com/M30-321-Ign/TextRecognition>

исходники, файлы и инсталятор.