

iContainersRead



Введение

Назначение: распознавание номеров контейнеров.

Версия: 1.0

Платформа: Windows 10, 11

Внешние зависимости

Необходимые пакеты и ссылки на них (ссылки с течением времени могут меняться)

1. Visual Studio 2019 redist (распространяемый пакет):
https://aka.ms/vs/17/release/vc_redist.x64.exe
2. OpenCV 4.7 – идет вместе с библиотекой
3. OpenVINO 2024.6.0 – идет вместе с библиотекой

Структуры

Описаны в заголовочном файле iContainers.h

iContainersSetting

Структура описывает настройки распознавания.

```
struct iContainersSetting
{
    bool correctSymbol = false;
    bool onlyFormatNumber = false;
};
```

Параметры:

correctSymbol – Использовать корректировочный символ по стандарту ГОСТ Р 52524-2019 (ИСО 6346:1995). Если символ не соответствует, то номер не будет возвращаться.

onlyFormatNumber – Проверка правильности формата номера - символы и цифры на своих местах.

NumberContainerResult

Структура, описывающая возвращаемый номер.

```
struct NumberContainerResult
{
    cv::Rect rect;
    std::string UID;
    std::string additionalCode;
    float confidence;
    float score;
};
```

Параметры:

rect – область детектированного номера;

UID – уникальный номер контейнера - 11 символов;

additionalCode – дополнительный код, если есть, если нет, то пустая строка;

confidence – вероятность номера при распознавании символов;

score – вероятность номера при детектировании.

Доступные функции

Описаны в заголовочном файле iContainers.h

iContainersInit

Инициализация объекта iContainersObject, включает в себя загрузку и инициализацию моделей.

```
extern "C" iContainersObject iContainersInit(
    char* detectionModelPath,
    char* symbDetectionModelPath,
    char* classificationModelPath,
    char* recognitionModelPath,
    char* license,
    int* error
);
```

Аргументы:

detectionModelPath – путь до модели детектирования номера;

symbDetectionModelPath – путь до модели детектирования символов;

classificationModelPath – путь до модели классификации символов;

recognitionModelPath – путь до модели распознавания строки;

license – лицензия представленная в виде строки символов;

error – информация об ошибке.

Функция возвращает объект iContainersObject или NULL в случае ошибки. Для просмотра кода ошибки необходимо передать указатель на целое число error.

iContainersRelease и iContainersReleaseP

Удаление объекта iContainersObject, включая удаление моделей из памяти.

```
extern "C" void iContainersRelease(
    iContainersObject * object
);
```

```
extern "C" bool iContainersReleaseP(  
    iContainersObject object  
);
```

Аргументы:

object – для **iContainersRelease** является указателем iContainersObject, для **iContainersReleaseP** передается без дополнительного указателя и возвращает BOOL.

iContainersSettings

Установить настройки распознавания.

```
extern "C" bool iContainersSettings(  
    iContainersObject object,  
    iContainersSetting settings  
);
```

Аргументы:

object – объект iContainersObject;

settings – настройки, заданные структурой iContainersSetting.

Функция возвращает true если настройки удалось применить, и false, если не удалось (object == NULL).

iContainersSettingsJSON

Установить настройки распознавания с помощью строки JSON.

```
extern "C" bool iContainersSettingsJSON(  
    iContainersObject object,  
    char* json  
);
```

Аргументы:

object – объект iContainersObject;

json – настройки в формате json. Пример:

```
{ "correctSymbol":1, "onlyFormatNumber":0 }
```

Функция возвращает true если настройки удалось применить, и false, если не удалось.

iContainersRead

Передать пакет изображений на распознавание и распознать номера.

```
extern "C" int iContainersRead(  
    iContainersObject object,  
    std::vector<cv::Mat> images  
);
```

Аргументы:

object – объект iContainersObject;

images – вектор 24 битных изображений (8UC3).

Возвращает 0 (iContErrors:: IC_OK) при удачной выполненной операции, в противном случае – код ошибки.

iContainersReadOnlyNumber

Передать пакет изображений только номеров на распознавание и распознать номера.

```
extern "C" int iContainersReadOnlyNumber(  
    iContainersObject object,  
    std::vector<std::vector<cv::Mat>> images  
);
```

Аргументы:

object – объект iContainersObject;

images – вектор векторов 24 битных изображений (8UC3).

Возвращает 0 (iContErrors:: IC_OK) при удачной выполненной операции, в противном случае – код ошибки.

iContainerAddImage

Добавить закодированное изображение - массив байт BMP, JPEG или PNG. Применяется в качестве альтернатив Read функций для не C++ программ. После этой функции нужно вызвать inference функцию.

```
extern "C" int iContainerAddImage(  
    iContainersObject object,  
    const char* imageBytes,  
    long sizeImage  
);
```

Аргументы:

object – объект iContainersObject;

imageBytes - массив байт, представляющий формат BMP, JPEG, PNG или TIFF;

sizeImage - размер массива.

Возвращает 0 (iContErrors:: IC_OK) при удачной выполненной операции, в противном случае – код ошибки.

iContainerInference

Распознать добавленные iContainerAddImage изображения.

```
extern "C" int iContainerInference(  
    iContainersObject object  
);
```

Аргументы:

object – объект iContainersObject.

Возвращает 0 (iContErrors:: IC_OK) при удачной выполненной операции, в противном случае – код ошибки.

iContainerInferenceOnlyNumber

Распознать добавленные iContainerAddImage изображения, считая, что в них четко ограниченный номер.

```
extern "C" int iContainerInferenceOnlyNumber(  
    iContainersObject object  
);
```

Аргументы:

object – объект iContainersObject.

Возвращает 0 (iContErrors:: IC_OK) при удачной выполненной операции, в противном случае – код ошибки.

iContainersGetResult

Получить результат распознавания текущего кадра.

```
std::vector<std::vector<NumberContainerResult>> iContainersGetResult(  
    iContainersObject object  
);
```

Аргументы:

object – объект iANRCRObject;

Возвращает вектор векторов номеров (может быть несколько номеров в кадре).

iContainersGetResultJSON

Получить результат распознавания текущего кадра в формате JSON.

```
extern "C" int iContainersGetResultJSON(  
    iContainersObject object,
```

```
char* outBuf,  
int sizeBuf  
);
```

Аргументы:
object – объект iANRCRObject;
outBuf – выходной буфера для JSON;
sizeBuf – размер выходного буфера.
Возвращает 0 и результат распознавания в формате JSON или пустую строку если возникла ошибка или ничего не нашлось. В противном случае - код ошибки.

iContainersVersion
Получить информацию о версии в виде строки символов. Пример:
iCONTPY:1.0.0.0.CPU

```
extern "C" bool iContainersVersion(  
char* outBuf,  
int sizeBuf  
);
```

Аргументы:
outBuf – выходной буфера для строки;
sizeBuf – размер выходного буфера.

Возвращаемые ошибки

Описаны в заголовочном файле **iContainersErrors.h**.

Табл.1. Ошибки

IC_OBJECTNULL = -1	Пустой объект iANRCRObject
IC_OK = 0	Операция успешна
IC_TRTEENGINELOADERROR = 1	Невозможно загрузить движок TensorRT
IC_COPYTOGPUMEMERROR = 2	Ошибка копирования в память GPU
IC_INFERENCEFAIL = 3	Ошибка при распознавании (инференс) изображений
IC_GETFROMGPUERROR = 4	Ошибка получения результатов распознавания с GPU
IC_NOIMAGES = 6	Нет изображений для распознавания
IC_IMAGENOT8U = 7	Изображения не соответствуют типу 24 бита.
IC_LICENSEERROR = 10	Ошибка лицензии
IC_ADDIMGCANNTDECODE = 11	Ошибка декодирования буфера – это не изображение
IC_ADDIMGNOTEQUALBATCH = 13	Ошибка – количество изображений в буфере слишком много
IC_OUTJSONBUFFERSMALL = 14	Выходной буфер слишком мал
IC_OPENVINOLOADERROR = 15	Невозможно загрузить модель OpenVINO