



DOCUMENT QA SDK

Developers Guide

[Content description](#)

This is a reference manual and configuration guide for the NeoCheck Document QA SDK product. It shows how to interact with the multiplatform libraries in order to analyse identity document images and extract quality parameters and card area for further processing.

Madrid, January 11th 2019

Disclaimer

The information contained within this document is and shall remain the property of NeoCheck. It must not be produced in whole or in part, or given or communicated to any third party without the prior consent of NeoCheck.

Whilst careful attention have been paid to ensure the appropriate referencing of information and sources, it is possible that a document of this nature might contain some errors. In such cases, upon notification NeoCheck will immediately analyze and make any required corrections or amendments.

©NeoCheck 2019



Release Notes

Version	Date	Description
1.0	11/01/2019	Initial Version
1.1	31/01/2019	Adjustments with glare
1.2	12/03/2019	Android Library
1.3	08/08/2019	Errors with big images fixed
1.4	09/08/2019	Document type classifier added



Index

Release Notes	2
1 Introduction	4
2 Installation	5
2.1 Document QA SDK Licensing.....	5
3 .NET Reference.....	6
3.1 Classes.....	6
3.2 Enums.....	6
3.3 Data Types.....	6
3.4 Methods.....	6
3.4.1 DocumentQASDK.GetImageQualityAsync method.....	7
3.4.2 DocumentQASDK.GetDocumentTypeAsync method.....	7
3.4.3 DocumentQASDK.CropDocumentAsync method	7
4 Android Reference	8
4.1 Classes.....	8
4.2 Data Types.....	8
4.3 UI Components	9
4.4 Engine Methods	9
4.4.1 GetVersionString method	9
4.4.2 Analyse method	9
4.4.3 Capture method	9
4.4.4 DetectCroppingPts method	10
4.4.5 CropTransform method	10
4.4.6 ClassifyFormat method	10
4.5 CroppableImageView Methods	11
4.5.1 GetCroppingPoints method	11
4.5.2 OnTouchEvent method	11
4.6 Camera2BasicFragment Methods.....	11
4.6.1 NewInstance method.....	11
4.6.2 SetListener method.....	11
4.7 CameraFragmentListener Interface	12
4.7.1 OnImageCaptured method	12
5 NeoCheck QA Android SDK Usage Recommendations	13



1 Introduction

This user manual describes the different functions and features provided by NeoCheck Document QA SDK. The purpose of this SDK is to provide functionality to recognize and analyse identity document pictures taken from webcam or mobile devices and extract quality parameters, as well as card area, for further verification processing. The SDK is designed for simple integration of identity document recognition into applications. The SDK has a high-level API that provides functionality for the identity document QA operations.

It is based on the following article that describes ICAO recommendations for MRTD documents:

<https://www.neocheck.com/mrtd-documents-capture-recommendations>

Features:

Document quality

Document quality is a process to analyze identity document and detect glare areas (due to an excessive light exposure or presence of camera flash light) or blur areas (unfocused image). The purpose of this process is to determine if the image meets the requirements needed for a further process to extract and verify identity document data.

Document classification

Document classification is a process to analyze identity document images and detect document format (passport, idcard, driver license, etc). Knowing this information helps user to determine whether the identity card has one or two sides, and/or show custom instructions to final user about the document capture process.

Document cropping

Document cropping is a process to detect and extract identity document area from a larger image, reducing noise and ensuring a better performance in a further process to extract and verify identity document data.



2 Installation

The Document QA SDK supports multiplatform environments, so several binary sets are provided to allow users to integrate it in the desired platform. There are some prerequisites per platform:

Windows platforms:

- Windows XP/7/10
- .NET framework 4.6

Android platforms:

- Min SDK Version: 21 (Lollipop 5.0)

2.1 Document QA SDK Licensing

The Document QA SDK is distributed under NeoCheck license, so to use the software it is required to contact your sales representative. This license will be checked regularly against the NeoCheck License Server to verify its validity, so periodical internet connection is also a prerequisite.



3 .NET Reference

NeoCheck Document QA SDK .NET libraries are a set of components for identify and processing identity document images. A list of all components and public methods are listed below.

3.1 Classes

NAME	DESCRIPTION
DocumentQASDK	The main entry point of the Document QA SDK. Provides methods for identity document image processing.

3.2 Enums

NAME	DESCRIPTION
DocumentType	The identity document classification format. Values: <ul style="list-style-type: none"> • NotSet: Unclassified document type • TD1: Most Id Cards, Voting cards and Driver License cards • TD2: Some Id Cards, such as French and old German Id Cards • TD3: Passports and most visas

3.3 Data Types

NAME	DESCRIPTION
DocumentTypeResult	Data result form document type classification, with properties: <ul style="list-style-type: none"> • DocumentType: DocumentType enum value (See chapter 3.2) • Probability: Confidence of the document type classification.
ImageQualityResult	Data result for image quality analysis, with properties: <ul style="list-style-type: none"> • IsBlurry: Check if the image is focused. • IsReadable: Check if the document is legible or not. • HasGlare: Check if there are regions with high brightness, due to excessive light exposure or camera flash. • DocumentType: DocumentTypeResult data with document type classification (if performed).
DocumentCroppingResult	Data result for document cropping, with properties: <ul style="list-style-type: none"> • CroppingPoints: List of points where the document borders were detected. • DocumentCropped: Image of the cropped document, using the cropping points.

3.4 Methods

The following is a list of all Document QA SDK public methods

3.4.1 DocumentQASDK.GetImageQualityAsync method

This method is called with an image where the identity document is present, and returns the image quality parameters in an object of type *ImageQualityResult* (See chapter 3.3)

PARAMETER	DESCRIPTION
Bitmap image	The image on which to perform the document quality analysis
bool detectDocumentType	(Optional) Determine whether or not to include the document type classification in the quality analysis. Default value is false

3.4.2 DocumentQASDK.GetDocumentTypeAsync method

This method is called with an image where the identity document is present, and returns the document type classification in an object of type *DocumentTypeResult* (See chapter 3.3)

PARAMETER	DESCRIPTION
Bitmap image	The image on which to perform the document type classification.

3.4.3 DocumentQASDK.CropDocumentAsync method

This method is called with an image where the identity document is present, and returns the document type classification in an object of type *DocumentCroppingResult* (See chapter 3.3)

PARAMETER	DESCRIPTION
Bitmap image	The image on which to perform the document cropping.
bool includeCroppedImage	(Optional) Determine whether or not to include the cropped document image in the result object. Default value is true.

4 Android Reference

NeoCheck Document QA SDK Android libraries are a set of components for identify and processing identity document images. A list of all components and public methods are listed below.

4.1 Classes

NAME	DESCRIPTION
com.neocheck.qa.android.Engine	The main entry point of the Document QA SDK. Provides methods for identity document image processing.

4.2 Data Types

NAME	DESCRIPTION
com.neocheck.qa.android.QAResult	<p>Represents the data structure that results from the QA analysis. Public attributes:</p> <ul style="list-style-type: none"> • int nGlareBlobs: the number of glare blobs detected in the image. This should be 0 for considering the image ok. • double td1Prob: the probability of the document to be a TD1. • double td2Prob: the probability of the document to be a TD2. • double td3Prob: the probability of the document to be a TD3. • boolean ok: whether or not the image passes the QA tests. • boolean isBlurry: whether or not the image is blurry or not focused. • android.graphics.RectF mrzCorners: the mrz corners when detected. If not detected, this is null.
com.neocheck.qa.android.CroppingConfig	<p>Holds the information related to the detected document cropping points. Public attributes:</p> <ul style="list-style-type: none"> • int x1: the x coordinate of the first cropping point. • int y1: the y coordinate of the first cropping point. • int x2: the x coordinate of the second cropping point. • int y2: the y coordinate of the second cropping point. • int x3: the x coordinate of the third cropping point. • int y3: the y coordinate of the third cropping point. • int x4: the x coordinate of the fourth cropping point. • int y4: the y coordinate of the fourth cropping point. • int width: the width of the base image from which the points were obtained. • int height: the height of the base image from which the points were obtained.

4.3 UI Components

NAME	DESCRIPTION
com.neocheck.qa.android.CroppableImageView	Handy UI component for allowing easy integration of manual cropping functionality
com.neocheck.qa.android.camera.AutoFitTextureView	View responsible of rendering the camera preview frames
com.neocheck.qa.android.camera.Camera2BasicFragment	Convenient class for orchestrating the functional integration of the key components in the document capture process.

4.4 com.neocheck.qa.android.Engine Methods

The following is a list of all Document QA SDK public methods

4.4.1 GetVersionString method

Returns the current version of the NeoCheck QA SDK. The version consists of three numbers separated by dots that are the major version, the minor version and the patch version respectively.

4.4.2 Analyse method

Performs the QA analysis on a defined region of a given image. This method returns a *QAResult* type

PARAMETER	DESCRIPTION
int rows	the number of rows of the image to process.
int cols	the number of columns of the image to process.
double regionAspectRatio	the aspect ratio of the centered region of the picture over which to do the processing.
double regionMaxSideRatio	the size relation between the region to process and the view that renders the camera preview.
int sensorOrientation	the angle in which the camera image should be rotated in order to match the device's native orientation.
boolean format	whether or not to infer the document format (TD1, TD2 or TD3).
java.nio.ByteBuffer grayscaleData	the bytearray containing the raw image grayscale data.

4.4.3 Capture method

Takes a picture of the actual camera frame and saves it to a specified local file path. If the capture is successful, it returns 0.

PARAMETER	DESCRIPTION
int rows	the number of rows of the image to process.
int cols	the number of columns of the image to process.



double regionAspectRatio	the aspect ratio of the centered region of the picture over which to do the processing.
double regionMaxSideRatio	the size relation between the region to process and the view that renders the camera preview.
int sensorOrientation	the angle in which the camera image should be rotated in order to match the device's native orientation.
java.nio.ByteBuffer grayscaleData	the bytearray containing the raw image grayscale data.
java.nio.ByteBuffer uvData	the bytearray containing the raw image uv data (from YUV format).
java.lang.String outputFilename	the output path in which to save the full-resolution captured image.
java.lang.String outputThumbnailFilename	the output path in which to save the thumbnail version of the captured image.

4.4.4 DetectCroppingPts method

Attempts to detect the cropping points of the document in the given image. Returns *CroppingConfig*:

PARAMETER	DESCRIPTION
int rows	the number of rows of the image to process.
int cols	the number of columns of the image to process.
double regionAspectRatio	the aspect ratio of the centered region of the picture over which to do the processing.
double regionMaxSideRatio	the size relation between the region to process and the view that renders the camera preview.
int sensorOrientation	the angle in which the camera image should be rotated in order to match the device's native orientation.
java.nio.ByteBuffer grayscaleData	the bytearray containing the raw image grayscale data.

4.4.5 CropTransform method

Applies the cropping to an image. If the cropping was successful, returns 0.

PARAMETER	DESCRIPTION
java.lang.String inputFilepath	The file path of the image from which to perform the cropping.
java.lang.String outputFilepath	The file path on which to write the cropped image.
int[] croppingPoints	The array containing the cropping point coordinates. The array should have the following shape [x1, y1, ..., x4, y4].

4.4.6 ClassifyFormat method

Predicts the format of a document from its cropped image.

PARAMETER	DESCRIPTION
java.lang.String croppedImgFilepath	The file path of the document cropped image from which to infer the format.

4.5 CroppableImageView Methods

Public attributes:

- **int[] xs**: the x coordinates of the cropping points.
- **int[] ys**: the y coordinates of the cropping points.
- **com.neocheck.qa.android.CroppingConfig**: the current cropping config.

4.5.1 GetCroppingPoints method

Returns the scaled cropping points as a convenient int array with the shape [x1, y1,... , x4, y4] according to a target image size.

PARAMETER	DESCRIPTION
android.util.Size imgSize	the size of the output image to which scale the cropping points to match.

4.5.2 OnTouchEvent method

Handles the touch events on the image view for allowing the user to adjust the cropping points.

PARAMETER	DESCRIPTION
android.view.MotionEvent event	the touch event to handle.

4.6 Camera2BasicFragment Methods

4.6.1 NewInstance method

Creates an instance of the class.

4.6.2 SetListener method

Sets the listener for handling the class delegated events.

PARAMETER	DESCRIPTION
com.neocheck.qa.android.camera.Camera2BasicFragment. NeoCheckCameraFragmentListener listener	the listener object to delegate events



4.7 CameraFragmentListener Interface

4.7.1 OnImageCaptured method

Callback for when an image has been captured.

PARAMETER	DESCRIPTION
java.lang.String fullImgFilepath	the file path to which the full-resolution captured image was written.
java.lang.String thumbnailFilepath	the file path to which the thumbnail version of the captured image was written.
com.neocheck.qa.android.CroppingConfig croppingConfig	the cropping config obtained from the automatic cropping tool.

5 NeoCheck QA Android SDK Usage Recommendations

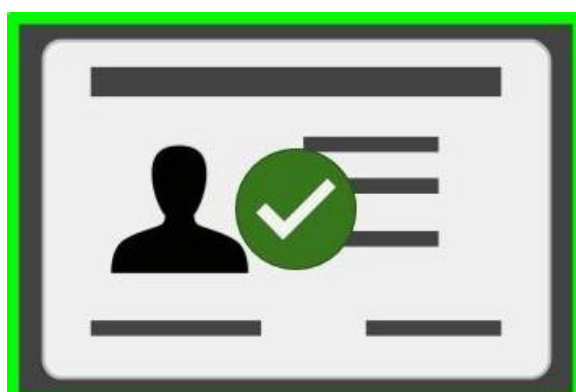
For best results, we recommend the users to follow the following tips:



The document should have high contrast with the background



The document should be contained inside the rectangular region



The document shouldn't be rotated too much