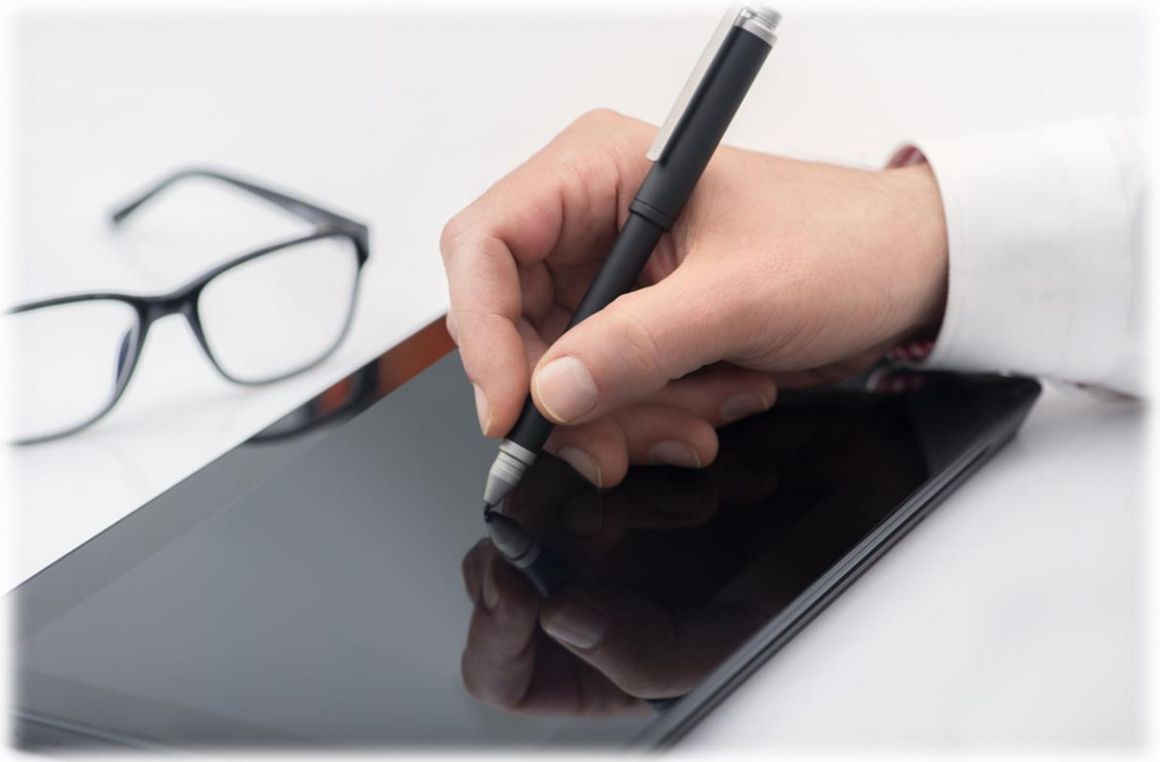


SIGNATURECHECK SDK

Developers Guide

Content description

This is a reference manual and configuration guide for the NeoCheck SignatureCheck SDK product. It shows how to interact with the .Net library in order to manipulate supported capture devices, crop signatures, extract pressure characteristics and match different samples in a fast and easy way.

Madrid, January 11th 2018

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 analyse and make any required corrections or amendments.

Release Notes

Version	Date	Description
1.0	11/01/2018	Initial Version

Index

1 Introduction 3
2 Installation 4
2.1 SignatureCheck SDK Licensing 4
3 .NET Reference 5
3.1 Classes 5
3.2 Events 5
3.3 Enums 5
3.4 EventArgs 5
3.5 Data Types 6
3.6 Methods 6
3.6.1 SignatureSDK.GetInstance method 6
3.6.2 SignatureSDK.Connect method 6
3.6.3 SignatureSDK.Disconnect method 6
3.6.4 SignatureSDK.Enable method 6
3.6.5 SignatureSDK.Disable method 6
3.6.6 SignatureSDK.Clear method 6
3.6.7 SignatureSDK.StartCapture method 7
3.6.8 SignatureSDK.AbortCapture method 7
3.6.9 SignatureSDK.Match method 7

1 Introduction

This user manual describes the different functions and features provided by NeoCheck Signature SDK. The purpose of this SDK is to provide functionality to manage signature pads and perform signature capture. The SDK is designed for simple integration into applications, which need biometric identity verification. The SDK has a high-level API that provides functionality for operate with signature pads

Features:

Signature Capture

Signature capture is a process when a signature pad connected to the computer is started and subject is requested to sign into a given area displayed in the signature pad. After signing, signature is extracted as image, along with metadata information (pressure matrix or others, depending on the specific signature pad hardware capabilities).

Signature Matching

Signature matching is a process when a signature can be identified by evaluating his biometric features and comparing them with another signature.

2 Installation

There are a set of prerequisites before start using the Signature SDK:

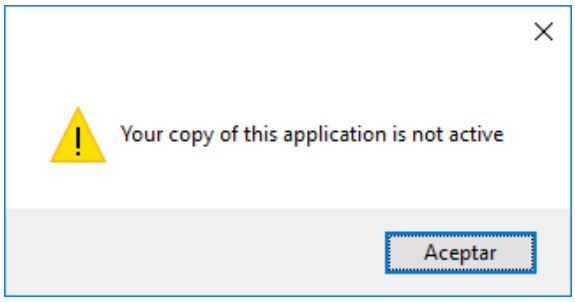
- Windows XP/7/10
- .NET framework 4.6

The Signature SDK comes as a set of DLL libraries that are distributed together, along with a Demo application and C# sample code.

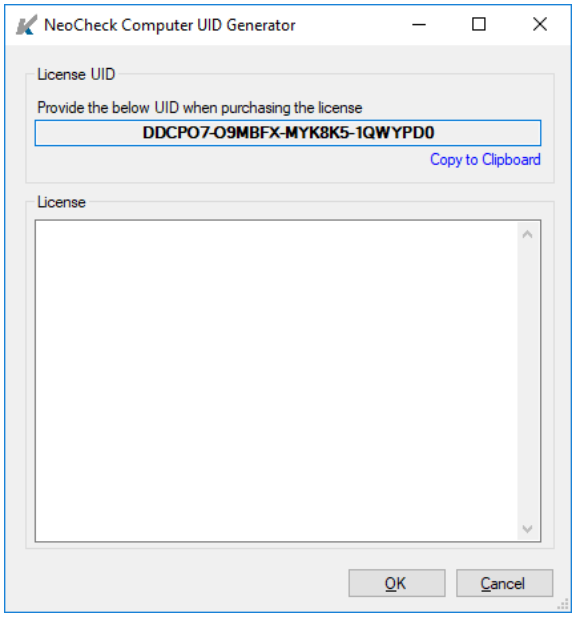
2.1 SignatureCheck SDK Licensing

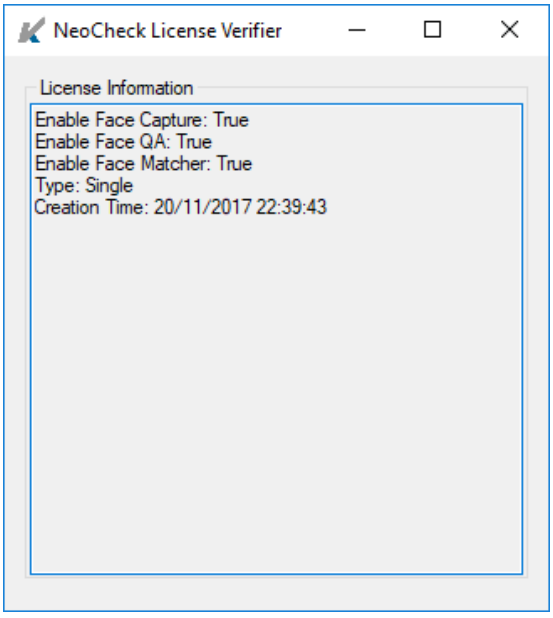
The Signature SDK uses a third-party software provided to perform signature 1:1 matching. The software requires to obtain a license on this component.

In order to activate the SDK, it is necessary to execute the NeoCheckLicenseVerifier tool. The following message will be displayed until a valid NeoCheckLicense.lic file is found:



A new window will be loaded showing your machine UID which should be sent to licenses@neocheck.com for generating the activation code. Once you paste the provided code in the lower textbox, a .lic file is going to be created in the execution route.





3 .NET Reference

NeoCheck Signature SDK has a set of components for working with signature pads connected to the computer, and performing signature capture.

3.1 Classes

NAME	DESCRIPTION
SignatureSDK	The main point of entry of the Signature SDK. Provides methods and events for working with pads and performing signature capture.
GenericSignaturePad	Represents the base class to operate with signature pads connected to the computer, providing methods to capture signature.
EPadLinkSignaturePad	ePadLink signature pad implementation.
TopazSignaturePad	Topaz signature pad implementation.
ISignatureMatcher	Represents the base class to perform signature matching.
FlorentisSignatureMatcher	Florentis signature matcher implementation.

3.2 Events

NAME	DESCRIPTION
SignatureCaptureCompleted	Event raised by a signature pad implementation when a signature capture process has finished. It includes the signature image in the event argument, and the global result of the capture process.

3.3 Enums

NAME	DESCRIPTION
SignaturePadImplementation	The implementations available to interact with signature pads connected to the computer. Values: Topaz EPadLink
SignatureMatcherImplementation	The signature matcher implementations available. Values: Florentis
SignatureCaptureStatus	The signature status result when performing a matching operation. Values: NotStarted Finished SignatureNotFound Canceled

3.4 EventArgs

NAME	DESCRIPTION
SignatureCaptureCompletedEventArgs	Event raised by the signature pad implementation when performing a signature capture operation. It contains the signature image and the global result of the capture operation. See SignatureCaptureStatus in Chapter 2.3.

3.5 Data Types

NAME	DESCRIPTION
Signature	Information about the signature captured by a signature pad. It includes the signature Bitmap image

3.6 Methods

Following a list of all public methods from Signature SDK, pads and signature matcher implementations.

3.6.1 SignatureSDK.GetInstance method

Returns a unique instance (Singleton) of the Signature SDK.

PARAMETER	DESCRIPTION
string SignaturePadImplementation	The signature pad implementation used to interact with devices connected to the computer.

3.6.2 SignatureSDK.Connect method

This method initializes the signature pad.

3.6.3 SignatureSDK.Disconnect method

This method disconnects the signature pad, releasing all resources used.

3.6.4 SignatureSDK.Enable method

This method enables the signature pad for user input.

3.6.5 SignatureSDK.Disable method

This method disables the signature pad for user input.

3.6.6 SignatureSDK.Clear method

This method clears the signature pad screen.

3.6.7 SignatureSDK.StartCapture method

This method allows the client application to perform a signature capture process using the signature pad. It is an asynchronous process, limited in time by a timeout value that can be passed as parameter. Either if capture is successful or any error or timeout occur, the client application can be informed about the result of the capture process by subscribing to the event SignatureCaptureCompleted, described in Chapter 1.2.

PARAMETER	DESCRIPTION
Int timeout	The maximum time (in milliseconds) for performing a signature capture. The default value is 10000 (10 seconds), and a 0 value means infinite time.

3.6.8 SignatureSDK.AbortCapture method

This method aborts a running capture process in the signature pad. It raises a SignatureCaptureCompleted event when signature pad has aborted the capture process and a new one can be started.

3.6.9 SignatureSDK.Match method

Returns a value indicating the similarity of the signatures passed as parameters. The return value is a float between 0 and 1.

PARAMETER	DESCRIPTION
Bitmap signature1	The first image to perform the signature matching.
Bitmap signature2	The second image to perform the signature matching.

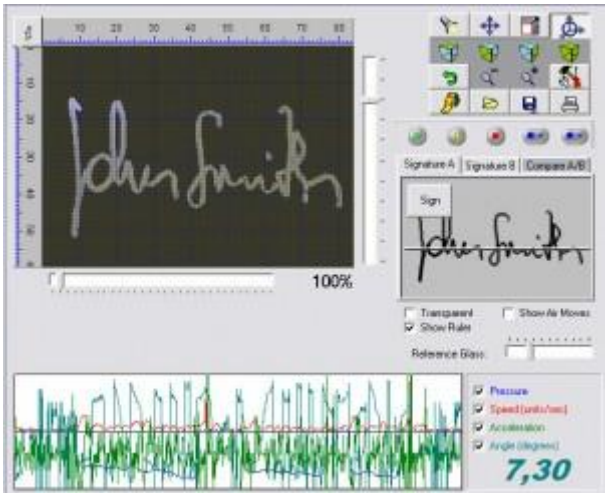


Figure 1 - Live trace signature parameters