

Build and Use Intel® Deep Learning a undertread undertread undertread undertread undertread undertread undertread undertread undertread undertread

I amount indefined undefined undefin

undermed underme

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visit www.intel.com/design/literature.htm.

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands and undefined un may be claimed as the property of others. ed undefined undef

defined undefined undefine Build and Use Intel® Deep Learning Streamer on the Windows* OS onitabnu house User Guide



		d undefine	defined
Aine	ed unde	Introduction	no.
ied unde Co	nten	ts undefine	
1.0		Introduction	5
	1.1	Terminology	6 define
	1.2	Reference Documents	
2.0	ind	Build Intel® Deep Learning Streamer on Windows* OS	
	2.1	System Setup	7
defil	2.2	Build Intel® Deep Learning Streamer Source	
, uno		2.2.1 Clone the Intel [®] Deep Learning Streamer source	
		2.2.2 Set up the Build Environment	
		2.2.3 Build the Intel® Deep Learning Streamer	9
		2.2.4 Install Built Files (Optional)	.10
3.0		Run Intel® Deep Learning Streamer on Windows* OS	11, dei
	3.1	Run Intel® Deep Learning Streamer	
	100	3.1.1 Set Environment and Download Models	
<u>م</u> ۵		3.1.2 Run samples	
defi	3.2	Use Intel® Deep Learning Streamer Library in New Projects	
4.0		Conclusion	15
inec		edu	-
		leftine ineo	
		uno-	713-
Fig	ures	ned to dunc	13 undefin
		afilition and a second se	20.

Figures

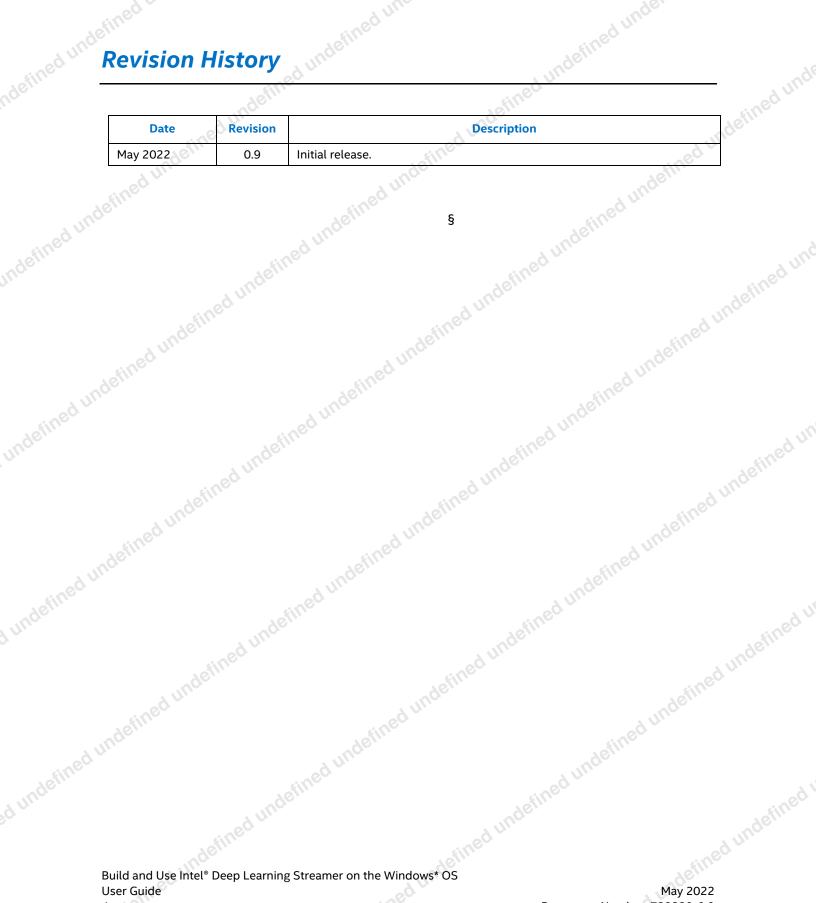
4.0	Conclusion	una		
	define	stined		ed un
Figures	ined une	ad under		ndefille
Figure 1.	Run Sample Result	dine		
Figure 2.	Snapshot for Adding Property Sheet	<u>)</u>		
Figure 3.	Property Sheet for GStreamer* Libraries			
nder	indefine	defineo		
Tables	defined u	sined unce		ed u
Table 1.	Terminology		6	define

rigundefined under **Tables**

ndefined	Tables	sined unoc	du	nden	
d unoc	Table 1.	Terminology	nodefiner	6	defined c
	Table 2.	Reference Documents	Letined b.	6	
	stined	01.	red unde	d undefinit	
ndefined	unde	undefi		adefinect	
undefine		defineo	stined	71.	ed l
		sined un	ed under		undefille
		Build	d and Use Intel® Deep Learning Strea	mer on the Windows* OS	

Build and Use Intel® Deep Learning Streamer on the Windows* OS beningbrin house User Guide - undefined 3

Revision History



Build and Use Intel® Deep Learning Streamer on the Windows* OS banitabnu hoose User Guide . G 4 unde 4 unde



1.0 Introduction

Intel® Deep Learning Streamer (Intel® DL Streamer) is a media streaming analytics framework that is based on the GStreamer* multimedia framework. Intel® DL Streamer is used to create complex media analytics pipelines. The Intel® DL Streamer's powerful and complex media analytics have new elements that can run inference operations using the Intel® Distribution of OpenVINO® Toolkit Inference Engine backend across Intel® architecture - CPU, iGPU and Intel® Movidius® VPU.

The Intel® DL Streamer facilitates media analytics through the following ways:

- Writes less code and gets better performance
- Quickly develops, optimizes, benchmarks, and deploys video and audio analytics pipelines in the Cloud and at the Edge
- Analyzes video and audio streams, creates actionable results, captures results, and sends them to the cloud
- Leverages the efficiency and computational power of Intel hardware platforms

For Linux* system, the Intel® DL Streamer is already integrated inside the Intel® Distribution of OpenVINO[™] toolkit. Users can use the Intel® DL Streamer after installing the OpenVINO[™] toolkit. The Intel® DL Streamer is not integrated inside the OpenVINO[™] toolkit for Windows* OS. Users on the Windows* OS will need to build the Intel® DL Streamer separately.

This guide provides the step-by-step instructions on how to build and use the Intel® DL Streamer on the Windows* 10 system.

> Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide 5

intel fined undefine

Introduction

Terminology

Terminology Table 1.

Terminology		defined un	4 unde
	Term	Description	defineo
	Intel [®] DL Streamer	Intel® Deep Learning Streamer	Inc
	OpenVINO™ Toolkit	Open Visual Inference & Neural Network Optimization Toolkit	

undefined undefined undefined

fined undefined undef

1.1

Reference Documents

Log in to the Resource and Documentation Center (rdc.intel.com) to search for and download the document numbers listed in the following table. Contact your Intel field representative for access.

Note: Third-party links are provided as a reference only. Intel does not control or audit thirdparty benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether the referenced data are accurate.

Table 2. **Reference Documents**

eference Documents Document	Document No./Location https://github.com/dlstreamer/dlstreamer/wiki §	
ntel® DL Streamer Wiki	https://github.com/dlstreamer/dlstreamer/wiki	IUC
ntel® DL Streamer Wiki	sined t	
	under a define	
	in the second se	
a under	odefili	
Lefined	ned un	
d unoc	indenti	tine
finec	ined L.	
	indefine	
	ed the ad unit	
Indeli	define	
	ed une	
Indell	define	nia
	ad une	

sined undefined undefined undefined undefined u Build and Use Intel® Deep Learning Streamer on the Windows* OS ered undefine User Guide unde

Build Intel® Deep Learning Streamer on Windows* OS

intel.undefined

2.0 Build Intel[®] Deep Learning Streamer on Windows* OS

To build the Intel[®] DL Streamer, you will need to install the dependency files first before setting up the environment to allow the make build system to find the dependencies.

You can follow the steps to finish the build.

System Setup

2.1

1) Windows 10* host system

This guide was tested on Windows 10 version 21H2*.

2) Install Python 3* for Windows* OS: https://www.python.org/downloads

This guide was tested on Python3.9*.

3) Install CMake 3.14* or newer for Windows OS*: https://cmake.org/download/

This guide was tested on CMake 3.14.2*.

 Install Visual Studio 2019* from the official site with C++ build tools: <u>https://visualstudio.microsoft.com/downloads/</u>

Visual Studio 2019* is required (previous versions such as Visual Studio 2017 will meet the minor compile error, but you will need to fix it manually).

5) Install GStreamer 1.16.2* runtime and development packages:

https://gstreamer.freedesktop.org/data/pkg/windows/1.16.2/gstreamer-1.0-msvcx86_64-1.16.2.msi

https://gstreamer.freedesktop.org/data/pkg/windows/1.16.2/gstreamer-1.0-develmsvc-x86_64-1.16.2.msi

GStreamer will be installed to C:\gstreamer\

Note: Intel recommends you to select "complete" install during the GStreamer* setup. Please check the installation guide for each component

6) Install pkg-config dependencies.

Download the following archives and extract them into the same folder: (e.g. C:work\pkg-config):

> Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide 7

intel ined under

Build Intel® Deep Learning Streamer on Windows* OS

- <u>https://download.gnome.org/binaries/win32/dependencies/pkg-config_0.26-</u> <u>1 win32.zip</u>
- <u>https://download.gnome.org/binaries/win32/dependencies/gettext-runtime_0.18.1.1-2_win32.zip</u>
- https://ftp.acc.umu.se/pub/GNOME/binaries/win32/glib/2.28/glib_2.28.8-<u>1_win32.zip</u>

Add the path inside the bin folder into the PATH environment variable. In this guide, this will be done later before the Intel® DL Streamer build.

You can also set the PATH environment to the system settings.

7) Install Intel[®] Distribution of OpenVINO[™] Toolkit 2021.4

The Intel® DL Streamer depends on the OpenVINO™ toolkit as inference backend.

Please register to download the latest version of <u>OpenVINO[™] toolkit for Windows*</u> <u>OS.</u>

Please download OpenVINO[™] toolkit version2021.4 that was tested in this guide.

For the installation process, Intel recommends you to follow the steps in the <u>OpenVINO[™] Toolkit Installation</u> Guide.

Please install the OpenVINO^m toolkit dependencies by following the steps in the OpenVINO^m toolkit installation guide.

Make sure to install OpenVINO[™] model_optimizer and model_downloader prerequisites. These tools will be used to download and convert the models needed.

```
@REM Install Model Optimizer prerequisites
cd C:\Program Files
(x86)\Intel\openvino_2021\deployment_tools\model_optimizer
\install_prerequisites
.\install_prerequisites.bat
```

@REM Install Open Model Zoo downloader requirements
pip3 install -r C:\Program Files
(x86)\Intel\openvino_2021\deployment_tools\open_model_zoo\
tools\downloader\requirements.in

Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide

Build Intel® Deep Learning Streamer on Windows* OS



Build Intel[®] Deep Learning Streamer Source

2.2.1

2.2

Clone the Intel[®] Deep Learning Streamer source

Clone the Intel[®] DL Streamer source repository into the working folder and check out the branch.

```
mkdir C:\work
cd C:\work
git clone -b preview/support-for-windows
https://github.com/openvinotoolkit/dlstreamer_gst.git
dlstreamer_gst
REM checkout the validated tag
git checkout 5aa67f0
```

Note: This guide uses "C:\work" as the working directory as an example. You can replace it with another directory if needed.

2.2.2

Set up the Build Environment

The environment settings only affect the current open command console. If you open a new console, you will need to set the environment variable again.

```
set PATH=%PATH%;C:\work\pkg-config\bin
set PATH=%PATH%;C:\gstreamer\1.0\x86_64\bin
set PKG_CONFIG_PATH=C:\gstreamer\1.0\x86_64\lib\pkgconfig
```

Build the Intel[®] Deep Learning Streamer

Create a build directory and generate the Visual studio solution and projects before building it.

```
cd C:\work\dlstreamer_gst\build
mkdir build
cd build
cmake -G "Visual Studio 16 2019" -A x64" .
@REM build
cmake --build . --config Release
```

The Intel® DL Streamer is now ready to use. For release version, the built files are under:

C:\work\dlstreamer\dlstreamer_gst\build\intel64\Release\bin\Release

May 2022 Document Number: 729229-0.9 Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide



Build Intel® Deep Learning Streamer on Windows* OS

Note: The Intel[®] DL Streamer needs to be built as a 64-bit since it depends on the OpenVINO[™] toolkit, which supports a 64-bit build.

2.2.4 Install Built Files (Optional)

You can install the built files to the GStreamer* library path for GStreamer to locate the files.

You can also use the built files in the Release folder by setting the GST_PLUGIN_PATH and pointing it to the Release folder's location.

In this guide, the installation of the Intel® DL Streamer library includes the files to the GStreamer* default installation path. However, you can install the built files in another location.

Specify the CMAKE_INSTALL_PREFIX variable to tell cmake the install location. Then, make the "INSTALL" target to start installing.

If you did not set the installation location for the steps mentioned in Section 2.3.3, cmake will use the default "Program files" as the installation path.

Hence, re-generate the Microsoft Visual Studio* projects with the specified installation location. Then, build then install it.

For example:

```
cmake -DCMAKE_INSTALL_PREFIX=C:\gstreamer\1.0\x86_64 "Visual
Studio 16 2019" -A x64" ..
cmake --build . --config Release
cmake --build . --target INSTALL -config Release
```

The Intel® DL Streamer is now installed to the default GStreamer* library directory: C:\gstreamer\1.0\x86_64\lib and included files are also installed in: C:\gstreamer\1.0\x86_64\include

Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide

Run Intel® Deep Learning Streamer on Windows* OS

intel undefine

3.0 Run Intel® Deep Learning Streamer on Windows* OS

Since the Intel® DL Streamer is an extension of GStreamer*, runing the Intel® DL Streamer on the Windows* OS is similar as running the GStreamer* applications. The Intel® DL Streamer added more elements with the "gva" prefix specifically for complex media analytics pipelines.

Depending on the Intel[®] DL Streamer libraries locations, you will need to set the path environment in order for GStreamer to find the elements. If the Intel[®] DL Streamer is already installed in the GStreamer installation directory, there are no more additional steps.

3.1 Run Intel[®] Deep Learning Streamer

3.1.1

Set Environment and Download Models

After Intel® DL Streamer is built, you can run a sample to check if it works.

You can find the samples in folder C:\work\dlstreamer_gst\win_samples. Before using the samples, use this script to download the models required for samples: download_models.bat (located in win_samples folder)

```
@REM set MODELS_PATH
set MODELS_PATH=C:\work\modles
```

@REM set OpenVINO environment again if you closed previous build console

C:\Program Files (x86)\Intel\openvino_2021\bin\setupvars.bat

```
@REM set Gstreamer PATH
set PATH=%PATH%;C:\gstreamer\1.0\x86 64\bin
```

@REM set DL Streamer library PATH
C:\work\dlstreamer_gst\win_scripts\setup_env.bat

```
@REM download the models now:
cd C:\work\dlstreamer_gst\win_samples
.\download models.bat
```

Note: The download_models.bat script needs a MODELS_PATH environment variable to save the downloaded models. The models need to be downloaded only once.

Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide

intel fined undefine

undefined undefined undefi

Run Intel® Deep Learning Streamer on Windows* OS

3.1.2 **Run samples**

Run the command line sample: 1)

> Command line sample is started by launching the gst-launch-1.0 console app. This is a tool that builds and runs basic GStreamer* pipelines.

```
@REM run the command line sample from script file
cd
C:\work\dlstreamer gst\win samples\gst launch\face detection
and classification
. \ face detection and classification.bat
```

fined undefined Jefined undefined un Note: The video path in the bat file is an online video from Github*. If you have connection issues, please download it manually into your local directory or select a local video.

Run cpp sample: 2)

undefined undefined undefined undefined

The cpp sample source files are under:

C:\work\dlstreamer gst\win samples\cpp\draw face attributes

It was already built during the Intel® DL Streamer build step.

@REM run the cpp command cd C:\work\dlstreamer gst\build\intel64\Release\bin\Release

draw face attributes.exe -i head-pose-face-detection-femaleand-male.mp4

el* De fined undefined undefined undefined undefined u Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide 12

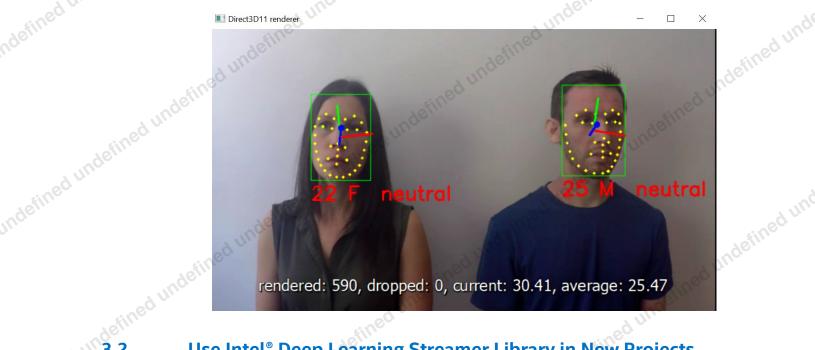
Run Intel® Deep Learning Streamer on Windows* OS

Figure 1. Run Sample Result



X

Direct3D11 renderer



3.2

Use Intel® Deep Learning Streamer Library in New Projects

If you want to use the Intel® DL Streamer in your own (cpp) projects, you will need the reference samples to write the code for adding the required header files and libraries.

You will need to select the 64-bit platform.

GStreamer* has provided a "Property Sheet" to include the dependencies easily. In Visual Studio 2019, the property manager can be found in View>Property Manager.

Right click on project name, select "Add Existing ...," then load the following props: "STREAMER ROOT X86%\share\vs\2010\libs and load gstreamer-1.0.props". stined undefined undefined u

indefined undefin Figure 2. Snapshot for Adding Property Sheet



May 2022 Document Number: 729229-0.9 Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide

intel fined undefine

ned undefine Run Intel® Deep Learning Streamer on Windows* OS

hed undefined undefined undefined undefined After this is done, the GSteramer* dependencies will be included in your project.

Figure 3. Property Sheet for GStreamer* Libraries

_			
	Property Manager - gobject-2.0	× ∓ €	dl_
undefined undefined unde	🔎 🎾 🔚 🛧 🦊 🗎	define	5
ed u.	🔺 🕵 dl_streamer_test	une	
efine	🔺 📄 Debug x64	neo	
nde	🖋 gstreamer-1.0	<i>.</i>	
du.	🔎 gobject-2.0		
stines	🔺 🔑 glib-2.0		
der	Common		
	Microsoft.Cpp.x64.		200
	Application	nno	
	🖉 🔑 Unicode Support	ed t	
6	🔑 Core Windows Libr	aries defined uno	
d Ull	Release x64	nde	
ineu	🔎 gstreamer-1.0	du	
4eth.	🔎 gobject-2.0	inec	
UNC	🔺 🔑 glib-2.0		
une ^O		· · · · · · · · · · · · · · · · · · ·	Loo de la 1
46f11.	The property sheet descript	ion above also works for	inte
undefined undefined und	This property sheet contains	s the directories where th	ie he
	ocated It also includes the		

The property sheet description above also works for Intel® DL Streamer.

This property sheet contains the directories where the headers and libraries are located. It also includes the necessary options for the compiler and linker, so you do not need to change anything else in your project.

Note: If you need to use OpenCV libraries, you will need to add them manually as they are not ya ele D. included in the property sheet above. undefined undefined undefined unde

fined undefined undefined undefined undefined u Build and Use Intel® Deep Learning Streamer on the Windows* OS User Guide 14 20





.cu ndefined undefined

Junderined underined underined underined underined underined underined underined underined underined underined

ndefined undefined undefin Build and Use Intel® Deep Learning Streamer on the Windows* OS e and undefined User Guide 15