Qualcomm Technologies, Inc.

Qualcomm Intelligent Multimedia Product Software Development Kit (QIMP SDK) Release Notes

80-70018-52 AA

April 9, 2025

Contents

1 QIMP SDK release 1.4

Contents

1.1 Release information

Table : Software version

Software	Version
Yocto	Scarthgap 5.0.6
Kernel	6.6.65

Table : Release tag version

Release tag	Version
Firmware release tag	r1.0_00075.0
Manifest release tag	qcom-6.6.65-QLI.1.4-Ver.1.1.xml
Meta-qcom-extras release tag	r1.0_00077.0
QIMP SDK release tag	qcom-6.6.65-QLI.1.4-Ver.1.1_qim-product-sdk-1.1.2.xml

Table : Supported platforms and reference kits

SoC platforms	Reference kits
QCS6490	• Qualcomm® RB3 Gen 2 Vision Development Kit • Qualcomm® RB3 Gen 2 Core Development Kit

SoC platforms	Reference kits
QCS5430	• Qualcomm® RB3 Gen 2 Lite Vision Development Kit • Qualcomm® RB3 Gen 2 Lite Core Development Kit
QCS9075	Qualcomm Dragonwing™ IQ-9075 Evaluation Kit (EVK)
QCS8275	Qualcomm® IQ8 Beta EVK

1.2 Contents of the release

The contents of the Qualcomm® Intelligent Multimedia Product (QIMP) SDK release include:

- Recipes for building the individual components:
  - Qualcomm® Intelligent Multimedia SDK (IM SDK)
  - Lite Runtime (LiteRT)

    Note: LiteRT was formerly known as TensorFlow Lite.

  - Qualcomm® Neural Processing SDK
  - Qualcomm® AI Engine direct SDK
- Sample applications that demonstrate how to use the Qualcomm IM SDK to develop AI edge-based applications.

To get started with the QIMP SDK, see Qualcomm Intelligent Multimedia Product (QIMP) SDK Quick Start Guide.

1.3 New features

The following new features applicable to QCS6490, QCS9075, and QCS8275 are introduced in the QIMP SDK release:

- Microservices:
  - Yolo V8-based person detection.
  - Face detection or recognition for access control.
  - General purpose daisy chain.

- AI-driven safety and monitoring system.
  - **Restricted zone alerts**: Define zones in the camera's FOV to generate alerts when someone enters these areas.
  - **PPE compliance**: Ensure safety gear compliance with alerts for missing or restricted items, configurable using a web interface.
- As part of the Yocto upgrade to Scarthgap, the GStreamer version has been upgraded to 1.22.
- Qualcomm plugins utilize a new color format, NV12_Q08C, which is introduced for NV12 with universal band width compression (UBWC). This color format replaces the previous method of using flags such as compression=ubwc.
- V4L2 supports `dmabuf` and `dmabuf-import` modes for efficient zero-copy data transfer.
- Audio AI: End-to-end support for general audio classification including sampling, preprocessing, inference, and post-processing.
- The docker base image is updated to Ubuntu 24.04 and the GStreamer version within the docker environment is 1.24.

For QCS9075:

- AI/ML parallel inference supports up to 32 streams.

1.4 Sample applications

Sample application	Description	Supported SoC
AI/ML applications		
`gst-ai-audio-classification`	Classification on streams from audio source.	QCS6490, QCS9075, and QCS8275
`gst-ai-metadata-parser-example`	Parses metadata using appsink plugin on streams from camera, file, or RTSP sources. Also, it provides the human count from the stream.	QCS6490, QCS9075, and QCS8275
`gst-ai-usb-camera-app`	USB single camera streaming for preview, video encoder, or network (RTSP) along with object detection and preview.	QCS6490, QCS9075, and QCS8275
Video application		

Sample application	Description	Supported SoC
`gst-opencv-transform`	Video playback using opencv API. Parses input video file, captures frame using cv video capture, and streams videoplay and composites to display using waylandsink.	QCS6490, QCS9075, and QCS8275
`gst-jpg-image-decode`	Decodes JPEG images where user can view decoded images on waylandsink.	QCS6490, QCS9075, and QCS8275
`gst-camera-opencv-resize`	Live camera stream using opencv API, which includes color conversion and resize. The application captures camera frame as an input using cv video capture, converts to RGBA using downstream plugin, and resizes to user–specified resolution. Resize output displays on screen.	QCS6490, QCS9075, and QCS8275
`gst-jpg-decode-example`	Collects the live video input from a camera, file, or an RTSP stream and uses the Qualcomm® Neural Network face detection model to produce a preview with the overlaid AI model output on the HDMI display.	QCS6490, QCS9075, and QCS8275

Note: `gst-usb-camera-example` is removed from multimedia sample applications. Use `gst-ai-usb-camera-app` instead of `gst-usb-camera-example`.

For the complete list of sample applications supported in the QIMP SDK and instructions on how to run them, see Sample applications.

1.5 Issues resolved

The following issues are resolved in the QIMP SDK release:

- A drop of 1 to 2 fps may be observed with the three-stream camera use case.

- Concurrent streams with smart codec functionality encounter cleanup errors at end of stream.
- Pose estimation for the `facemap_3dmm_quantized.bin` (Qualcomm Neural Network) AI hub model doesn't work with the Qualcomm IM SDK pipeline due to caps mismatch.
- Caps mismatch with the AI hub face detection Qualcomm Neural Network model.
- Support for the FastCV engine is disabled in qtivtransform.

1.6 Limitations

The following are the known limitations in the QIMP SDK release:

- When using the `qtioverlay` plugin with detection models, frame drops may occur, especially with many detections.

  Workaround: Use `qtivcomposer` for detection-based ML use cases.
- AI/ML parallel inference for 32 streams varies between 20–30 fps depending on the model and input stream, which is less than the expected 30 fps.
- Segfault occurs while using *Ctrl + C* for Qualcomm® Neural Processing use cases with DLC models.
- Frame drops are observed with the Qualcomm Neural Network plugin while running on GPU delegate.
- Low FPS is observed with GStreamer pipelines for daisychain and pose detection.
- Stability issues are observed with GStreamer pipeline in batched model use cases.
- High inference time is observed with `deeplabv3_resnet50.dlc`.
- Concurrent streams with smart codec functionality encounter pipeline stalls at the end of the stream.
- Reverse playback is limited to video streams with a GOP length that fits within the buffer limitations of the video driver (<= 26).
- Support for GStreamer daemon (GStd), NV12 UBWC compressed streams, and sample apps dependent on NV12 UBWC aren't enabled inside the docker.
- The `gst-ai-parallel-inference` sample application hangs for QCS8275.
- For QCS9075 and QCS8275:
  - A hang is observed while running use cases with batched models using GStreamer.
  - Random corruption is observed at the beginning of playback for NV12 UBWC compressed low-resolution streams (480p).

2 QIMP SDK release 1.3

Contents

2.1 Release information

Table : Software version

Software	Version
Yocto	Kirkstone 4.0.22
Kernel	6.6.52

Table : Release tag version

Release tag	Version
Firmware release tag	r1.0_00058.0
Manifest release tag	qcom-6.6.52-QLI.1.3-Ver.1.1
Meta-qcom-extras release tag	r1.0_00059.0
QIMP SDK release tag	qcom-6.6.52-QLI.1.3-Ver.1.1_qim-product-sdk-1.1.2

Table : Supported platforms and reference kits

SoC platforms	Reference kits
QCS6490	• Qualcomm® RB3 Gen 2 Vision Development Kit • Qualcomm® RB3 Gen 2 Core Development Kit

SoC platforms	Reference kits
QCS5430	• Qualcomm® RB3 Gen 2 Lite Vision Development Kit • Qualcomm® RB3 Gen 2 Lite Core Development Kit
QCS9075	Qualcomm Dragonwing™ IQ-9075 EVK
QCS8275	Qualcomm® IQ8 Beta Evaluation Kit

2.2 Contents of the release

The contents of the Qualcomm® Intelligent Multimedia Product (QIMP) SDK release include:

- Recipes for building the individual components:
  - Qualcomm IM SDK
  - LiteRT
  - Qualcomm Neural Processing SDK
  - Qualcomm AI Engine direct SDK
- Sample applications that demonstrate how to use the Qualcomm IM SDK to develop AI edge-based applications.

To get started with the QIMP SDK, see Qualcomm Intelligent Multimedia Product (QIMP) SDK Quick Start Guide.

2.3 New features

The following new features applicable to QCS6490, QCS9075, and QCS8275 are introduced in the QIMP SDK release:

- AI/ML supports execution of batched Qualcomm Neural Processing SDK models and batched Qualcomm Neural Network models.
- Container supports application deployment using an API-based approach–GSTD.

2.4 Sample applications

Sample application	Description	Supported SoCs
AI/ML applications		
`gst-ai-face-detection`	Collects the live video input from a camera, file, or an RTSP stream and uses the Qualcomm Neural Network face detection model to produce a preview with the overlaid AI model output on the HDMI display.	QCS6490, QCS9075, and QCS8275
`gst-ai-face-recognition`	Collects the live video input from a camera or an RTSP stream and shares this input for face detection and recognition.	QCS6490, QCS9075, and QCS8275
Video application		
`gst-ai-smartcodec-example`	Reduces the network bandwidth or storage for input feed from a camera or file source.	QCS6490, QCS9075, and QCS8275

For the complete list of sample applications supported in the QIMP SDK and instructions on how to run them, see Sample applications.

2.5 Python sample applications

Sample application	Description	Supported SoCs
AI/ML applications		
`gst-parallel-inference.py`	Supports multiple AI/ML models run in parallel (object detection, classification, segmentation, and pose detection) on streams from a camera, file source, or RTSP.	QCS6490, QCS9075, and QCS8275
`gst-daisychain-detection-pose.py`	Collects the live video input from a camera or an RTSP stream and shares this input for face detection and recognition.	QCS6490, QCS9075, and QCS8275
Video application		

Sample application	Description	Supported SoCs
`gst-concurrent-videoplay-composition.py`	Allows concurrent video playback for MP4 AVC (H.264) videos and performs composition on a video wall display.	QCS6490, QCS9075, and QCS8275
Camera application		
`gst-multi-camera-stream-example.py`	Enables you to stream from two camera sensors simultaneously.	QCS6490

For the complete list of Python sample applications supported in the QIMP SDK and instructions on how to run them, see Python applications.

2.6 Issues resolved

The following issues are resolved in the QIMP SDK release:

- 18 fps is observed with `gst-ai-parallel-inference` for file source.
- `gst-ai-parallel-inference` hangs and black screen is observed at EOS for RTSP source.
- RTSP sink streaming fails to play when you use multiple sinks such as RTSP, file sink, and display with two or more input streams in `multi-input-output-object-detection`.
- `gst-ai-monodepth` and `gst-ai-parallel-inference` fails on QCS9075.

2.7 Limitations

The following are the known limitations in the QIMP SDK release:

- When using the `qtioverlay` plugin with detection models, frame drops may occur, especially with many detections.

  Workaround: Use `qtivcomposer` for detection-based ML use cases.
- A drop of 1–2 fps may be observed with the three-stream camera use case.
- AI/ML parallel inference for 24 streams varies between 25–30 fps, which is less than the expected 30 fps.
- Segfault occurs while using *Ctrl + C* for Qualcomm® Neural Processing use cases with DLC models.
- Frame drops are observed with the Qualcomm Neural Network plugin while running on GPU

delegate.

- Low FPS with daisychain detection and pose GStreamer pipeline.
- Stability issues are observed with GStreamer pipeline in batched model use cases.
- High inference time is observed with `deeplabv3_resnet50.dlc`.
- Concurrent streams with smart codec functionality encounter pipeline stalls and cleanup errors at the end of the stream.
- Pose estimation for the `facemap_3dmm_quantized.bin` (Qualcomm Neural Network) AI hub model doesn't work with Qualcomm IM SDK pipeline due to caps mismatch.
- Caps mismatch with the AI hub face detection Qualcomm Neural Network model.
- The `gst-ai-parallel-inference` sample application hangs for QCS8275.
- Support for the FastCV engine is disabled in `qtivtransform`.

3 QIMP SDK release 1.2

3.1 Release information

Table : Software version

Software	Version
Yocto	Kirkstone 4.0.20
Kernel	6.6.38

Table : Release tag version

Release tag	Version
Firmware release tag	r1.0_00049.0
Release tag	qcom-6.6.38-QLI.1.2-Ver.1.1
Meta-qcom-extras release tag	r1.0_00050.0
QIMP SDK release tag	qcom-6.6.38-QLI.1.2-Ver.1.1_qim-product-sdk-1.1.2

Table : Supported platforms and reference kits

SoC platforms	Reference kits
QCS6490	• Qualcomm® RB3 Gen 2 Vision Development Kit • Qualcomm® RB3 Gen 2 Core Development Kit
QCS5430	• Qualcomm® RB3 Gen 2 Lite Vision Development Kit • Qualcomm® RB3 Gen 2 Lite Core Development Kit
QCS9075	Qualcomm Dragonwing™ IQ-9075 EVK

3.2 Contents of the release

The contents of the Qualcomm® Intelligent Multimedia Product (QIMP) SDK release include:

- Recipes for building the individual components:
  - Qualcomm IM SDK
  - LiteRT
  - Qualcomm Neural Processing SDK
  - Qualcomm AI Engine direct SDK
- Sample applications that demonstrate how to use the Qualcomm IM SDK to develop AI edge-based applications.

To get started with the QIMP SDK, see Qualcomm Intelligent Multimedia Product (QIMP) SDK Quick Start Guide.

3.3 New features

The following are the new features introduced in the QIMP SDK release:

- AI/ML supports:
  - QuickSRNet for video super resolution.
  - Running batched models through the LiteRT plugin, enabling more efficient processing of multiple inputs.
  - Enhanced workflow with stage-wise metadata, improving the clarity of the daisy chain pipeline.
  - Message queuing telemetry transport (MQTT) communication through dedicated plugins.
  - The Redissink plugin to submit ML metadata to the Redis database server.
- Video supports smart codec functionality, which dynamically adjusts bit rate, frame rate, and GOP based on the scene.
- Container supports:
  - Qualcomm IM SDK plugins and applications that allow you to use the same Qualcomm IM SDK applications in both docker and nondocker solutions.
  - Standalone LiteRT that lets you to have only LiteRT and use their own pipeline for the rest of the application.

  - Standalone Qualcomm Neural Processing for hardware acceleration of AI models, enabling you to have only Qualcomm Neural Processing and use their own pipeline for the rest of the application.
  - Python bindings on top of SNPE APIs and standalone Python applications that support reading offline data, preprocessing, inferencing, and postprocessing.

3.4 QCS9075 features

The following are the features introduced in the QIMP SDK release:

- AI/ML supports parallel inference for up to 24 streams.
- Video supports:
  - MJPEG video decoding and encoding sessions in CPU/GPU.
  - End-to-end 1080p concurrent video sessions for decode and encode:
    - 24 H264/H265 sessions decode
    - 24 H264 sessions encode
    - 12 H264 sessions decode and encode
  - 4K at 240 fps VP9 and AV1 decode.

3.5 Sample applications

Sample applications	Description	Supported SoCs
AI/ML applications		
`gst-ai-superresolution`	High resolution video frames from low-resolution input.	QCS6490
`gst-ai-multistream-batch-inference`	Batched AI inference (object detection and segmentation) from a file source.	QCS6490
`gst-ai-multistream-inference`	AI inference (object detection and classification) from multiple sources such as camera, file source, or RTSP.	QCS6490
`snpe_segmentation_app.py`	Segmentation using Python bindings from a docker.	QCS6490

Sample applications	Description	Supported SoCs
`gst-ai-classification`	Classification on streams from a camera, file source, or real-time streaming protocol (RTSP).	QCS9075
`gst-ai-object-detection`	Object detection on streams from a camera, file source, or RTSP.	QCS9075
`gst-ai-pose-detection`	Pose detection on streams from a camera, file source, or RTSP.	QCS9075
`gst-ai-segmentation`	Image segmentation on streams from a camera, file source, or RTSP.	QCS9075
`gst-ai-multi-input-output-object-detection`	Use cases for Edge AI boxes, which provide inputs through multiple sources such as camera, file source, or RTSP.	QCS9075
`gst-ai-daisychain-detection-classification`	Cascaded object detection and classification on images streamed from multiple sources such as camera, file source, or RTSP.	QCS9075
`gst-ai-daisychain-detection-pose`	Cascaded object detection and pose detection from multiple sources such as camera, file source, or RTSP.	QCS9075
`gst-ai-superresolution`	High resolution video frames from low-resolution input.	QCS9075
`gst-ai-multistream-inference`	AI inference (object detection and classification) from multiple sources such as camera, file source, or RTSP.	QCS9075
`gst-ai-multistream-batch-inference`	Batched AI inference (object detection and segmentation) from file source.	QCS9075
`snpe_segmentation_app.py`	Segmentation using Python bindings from a docker.	QCS9075
Camera applications		

Sample applications	Description	Supported SoCs
`gst-usb-single-camera-app`	USB single camera streaming for preview, video encoder, or network (RTSP).	QCS6490
`gst-camera-shdr-ldc-eis-example`	Super high dynamic range (sHDR), low dynamic range (LDR), and electronic image stabilization (EIS) camera imaging techniques.	QCS6490
`gst-smartcodec-example`	Reduces the network bandwidth or storage from camera input.	QCS6490 and QCS9075
Video applications		
`gst-concurrent-videoplay-composition`	Playback multiple videos composed on a device. Use cases where video wall can be used in retail spaces and digital signage.	QCS9075
`gst-videocodec-concurrent-playback`	Decoding and playback of multiple formats concurrently. Use cases for AI box and videoconferencing system, which sends video inputs in different formats.	QCS9075
`gst-video-transcode-example`	Video transcoding of AVC-to-HEVC or HEVC-to-AVC format.	QCS9075
`gst-transform-example`	Use cases where a video stream should be transformed (rotate, flip, and scale).	QCS9075

For the complete list of sample applications supported in the QIMP SDK and instructions on how to run them, see Sample applications.

3.6 Python sample applications

- `gst-camera-detect-display`, which processes camera stream to detect objects, overlay bounding boxes, and display the result.
- `gst-decode-detect-display`, which processes video stream to detect objects and display them with bounding boxes.
- `gst-camera-two-stream-detection-and-classification-side-by-side`, which detects and classifies objects from camera streams and displays the results side by

side.

- `gst-filesrc-2detection-classification-segmentation-side-by-side`, which processes video files to detect, classify, and segment objects and displays the results side by side.
- `gst-camera-two-stream-encode-file-detection-display`, which encodes camera streams, detects objects, and displays the result.
- `gst-camera-three-stream-encode-file-detection-display-classification-rtsp`, which encodes camera streams, displays detection results, and streams classification results over RTSP.
- `gst-rtspsrc-detection-display`, which decodes an RTSP stream, detects objects, overlays bounding boxes on them, and displays the result.
- `gst-camera-encode`, which records and encodes a camera stream.
- `gst-camera-rotate-downscale-file`, which processes a single camera stream by recording, downscaling, rotating, and encoding it.

3.7 Limitations

The following are the known limitations in the QIMP SDK release:

- When using the `qtioverlay` plugin with detection models, frame drops may occur, especially with many detections.

  Workaround: Use `qtivcomposer` for detection-based ML use cases.
- A drop of 1–2 fps may be observed with the three-stream camera use case.
- AI/ML parallel inference for 24 streams is achieved at 22 fps, which is less than the expected 30 fps.
- Segfault occurs while using *Ctrl + C* for Qualcomm® Neural Processing use cases with DLC models.
- Frame drops are observed with the Qualcomm Neural Network plugin while running on GPU delegate.
- Low FPS with daisychain detection and pose GStreamer pipeline.
- Stability issues are Observed with GStreamer pipeline in batched model use cases.
- High inference time is observed with `deeplabv3_resnet50.dlc`.
- 18 fps is observed with `gst-ai-parallel-inference` for file source.
- `gst-ai-parallel-inference` hangs and black screen is observed at EOS for RTSP source.

- RTSP sink streaming fails to play when you use multiple sinks such as RTSP, file sink, and display with two or more input streams in `multi-input-output-object-detection`.
- `gst-ai-monodepth` and `gst-ai-parallel-inference` fails on QCS9075.

4 QIMP SDK release 1.1

4.1 Release information

Table : Software version

Software	Version
Yocto	Kirkstone 4.0.18
Kernel	6.6.28

Table : Release tag version

Release tag	Version
Firmware release tag	r1.0_00039.2
Release tag	qcom-6.6.28-QLI.1.1-Ver.1.1.xml
Meta-qcom-extras release tag	r1.0_00040.0
QIMP SDK release tag	qcom-6.6.28-QLI.1.1-Ver.1.1_qim-product-sdk-1.1.3

Table : Supported platforms and reference kits

SoC platforms	Reference kits
QCS6490	• Qualcomm® RB3 Gen 2 Vision Development Kit • Qualcomm® RB3 Gen 2 Core Development Kit
QCS5430	• Qualcomm® RB3 Gen 2 Lite Vision Development Kit • Qualcomm® RB3 Gen 2 Lite Core Development Kit

4.2 Contents of the release

The contents of the QIMP SDK release include:

- Recipes for building the individual components:
  - Qualcomm IM SDK
  - LiteRT
  - Qualcomm Neural Processing SDK
  - Qualcomm AI Engine direct SDK
- Sample applications that demonstrate how to use the Qualcomm IM SDK to develop AI edge-based applications.

To get started with the QIMP SDK, see Qualcomm Intelligent Multimedia Product (QIMP) SDK Quick Start Guide.

4.3 New features

The following are the new features introduced in the QIMP SDK release:

- Camera supports:
  - Sample applications
    - `gst-add-streams-as-bundle-example`, which is a configuration of camera streams as a bundle, thus, saving on time.
    - `gst-camera-burst-capture-example`, which allows snapshot capture in the Burst mode.
    - `gst-camera-switch-example`, which allows to switch between two cameras.
    - `gst-add-remove-streams-runtime`, which allows the addition and removal of multiple streams dynamically.
  - Multiple streams with different framerate for each stream
  - USB camera using the v4l2src GStreamer plugin
  - Different video HDR modes: RAW linear SHDR (V2), YUV virtual channel SHDR (V3), and SHDR switch from linear to RAW linear SHDR (V2)
  - Electronic image stabilization (EIS) for preview only or preview + video use case
  - Reprocess to generate multiple streams from a single input stream based on configured

region of interest (ROI) coordinates

  - Continuous JPEG snapshots at 30 fps along with preview
- Video supports:
  - Playback framerate control
- Graphics supports:
  - User configurable polygon privacy mask and inverse privacy mask using the gstqtioverlay plugin
- AI/ML supports:
  - Sample applications
    - `gst-ai-monodepth`, which allows monodepth on images streamed from a camera.
    - `gst-ai-daisychain-detection-classification`, which allows cascaded object detection and classification on images streamed from a camera.
  - Multistage ML pipeline (daisy chaining) for advanced ML use cases; for example, detection followed by classification based on detected objects.
  - Socket-based gstqtisocketsrc and gstqtisocketsink plugins to support efficient/zero copy data flow between pipelines running inside docker and host.
  - Hardware-accelerated gstqtivsplit plugin to support input image crop based on ML detection ROI output. This plugin generates a stream per detected ROI that's used for multistage ML pipeline or other purposes like storage.
  - Concurrent AI pipelines without impacting the performance
  - Mono-depth estimation ML model (midas-v2) for depth estimation based on a single image

4.4 Sample applications

For a list of sample applications supported in the QIMP SDK and instructions on how to run them, see Sample applications.

4.5 Issues resolved

Fixed speed-based video playback.

4.6 Limitations

The following are the known limitations in the QIMP SDK release:

- When using the `qtioverlay` plugin with detection models, frame drops may occur, especially with many detections. Use qtivcomposer for detection-based ML use cases.
- A drop of 1–2 fps may be observed with the three-stream camera use case.

5 QIMP SDK release 1.0

5.1 Release information

Table : Software version

Software	Version
Yocto	Kirkstone 4.0.14
Kernel	6.6.17

Table : Release tag version

Release tag	Version
Firmware release tag	r1.0_00026.0
Release tag	qcom-6.6.17-QLI.1.0-Ver.1.4
Meta-qcom-extras release tag	r1.0_00028.0
QIMP SDK release tag	qcom-6.6.17-QLI.1.0-Ver.1.4_qim-product-sdk-1.1

Table : Supported platforms and reference kits

SoC platforms	Reference kits
QCS6490	• Qualcomm® RB3 Gen 2 Vision Development Kit • Qualcomm® RB3 Gen 2 Core Development Kit

5.2 Contents of the release

The contents of the QIMP SDK release include:

- Recipes for building the individual components:
  - Qualcomm IM SDK
  - LiteRT

  - Qualcomm Neural Processing SDK
  - Qualcomm AI Engine direct SDK
- Sample applications that demonstrate how to use the Qualcomm IM SDK to develop AI edge-based applications.

To get started with the QIMP SDK, see Qualcomm Intelligent Multimedia Product (QIMP) SDK Quick Start Guide.

5.3 New features

- Camera supports:
  - Sample applications
    - `gst-camera-single-stream-example`, which allows single camera streaming to display or video encoder, or network (RTSP).
    - `gst-multi-camera-example`, which allows concurrent streaming from two cameras.
    - `gst-multi-stream-example`, which allows single camera data duplicated into two streams using tee.
    - `gst-snapshot-stream-example`, which allows snapshot generation along with active video recording.
    - `gst-activate-deactivate-streams-runtime-example`, which allows camera resolution modifications on-the-fly without sensor restart.
    - `gst-appsink-example`, which is an application of transformations and other algorithms to a camera frame before encoding.
    - `gst-camera-metadata-example`, which is a metadata such as scene mode, white balance, and bit rate.
  - Real-time camera up to 4K at 30 fps or 1080p at 60 fps
  - Multistreams directly from a single camera
  - Multiple camera access from single or multiple client applications
  - Multiple snapshot support (RAW/JPEG) at configurable resolutions
  - Common camera controls such as contrast, international organization for standardization (ISO), and white balance are using the GStreamer camera plugin properties.
  - Camera metadata for fine-grain control of camera configurations.
- Video supports:
  - Sample applications

    - `gst-audio-video-encode`, which allows encoding of audio and video data, which is then stored into a user-specified output file.
    - `gst-concurrent-videoplay-composition`, which allows to playback multiple videos composed on a device.
    - `gst-videocodec-concurrent-playback`, which allows decoding and playback of multiple formats concurrently.
    - `gst-audio-video-playback`, which has audio and video playback capabilities.
    - `gst-video-transcode-example`, which allows video transcoding of AVC-to-HEVC or HEVC-to-AVC format.
    - `gst-video-playback-example`, which allows decoding and playback of a video file.
  - The file-based decode and playback feature
  - Video for Linux (V4L2)-based hardware encode and decode
  - 16 parallel instances for encode and decode
  - Encode supports:
    - Codecs: H.264, H.265
    - Controls: Profile, level, bit rate control, minimum and maximum quantization parameter (QP), group of pictures (GOP), entropy coding, slice, Hier-P, rotation, flip, and long term reference (LTR)
    - Dynamic property control: LTR, bit rate, and flip
    - 4K30 + 720p30 encode
  - Decode supports:
    - Formats: NV12, TP10 Qualcomm universal bandwidth compression
    - Playback controls: Play, pause, resume, forward, and reverse
    - Dynamic resolution change
    - Video graphics array (VGA at 640 x 480) with 16 parallel instances
    - 720p with 8 parallel instances
    - 1080p with 4 parallel instances
    - UHD with 2 parallel instances
- Audio supports:
  - Sample applications
    - `gst-audio-encode-example`, which allows audio recording.

    - `gst-audio-decode-example`, which allows playback of an audio file.
    - `gst-transform-example`, which has use cases where a video stream should be transformed (rotate, flip, and scale).
    - `gst-weston-composition-example`, which allows composition of various sources, specifically live camera input, and an offline file.
  - Hardware-accelerated audio capture and playback using pulse plugins
- Graphics supports:
  - Hardware-accelerated plugins for compose, transform, and conversion
  - Composition: N input to 1 output. For example: video wall, ML metadata, and picture-in-picture
  - Transform: Crop, rotation, scale (down/up), color convert, and flip
  - Split: 1 input to N outputs based on user input region of interest (ROI)
- AI/ML supports:
  - Sample applications
    - `gst-ai-classification`, which allows classification on a live camera stream.
    - `gst-ai-multi-input-output-object-detection`, which allows AI/ML inferencing on various input sources.
    - `gst-ai-object-detection`, which allows object detection on images streamed from a camera.
    - `gst-ai-parallel-inference`, which allows multiple AI/ML models running in parallel on a single live stream from a camera.
    - `gst-ai-pose-detection`, which allows pose detection on images streamed from a camera.
    - `gst-ai-segmentation`, which allows object segmentation on images streamed from a camera.
  - ML framework: LiteRT, Qualcomm Neural Processing SDK, and Qualcomm AI Engine direct SDK
  - Models: Classification, detection, pose estimation, and segmentation
  - Tensor-based input and output for fast processing
  - Submodule-based postprocessing for flexibility of adding support for new models
  - Overlay inference results on input frame for complete ML use case realization
- Miscellaneous includes:
  - Containerization reference

  - Socket source and sink plugins that:
    - Enable zero copy data path between two separate processes running their own GStreamer pipeline
    - Separate processes/applications are either within the native layer or within the docker execution environment

5.4 Sample applications

Sample applications are classified into two categories, each tailored to specific use cases like retail, AI edge box, IP camera, drones, and robotics.

- Multimedia applications: These focus on camera, video, audio, and graphics functionalities.
- AI/ML applications: These primarily focus on AI and ML capabilities.

For a list of sample applications supported in the QIMP SDK and instructions on how to run them, see Sample applications.

5.5 Limitations

The following are the known limitations in the QIMP SDK release:

- The FastCV engine of `qtimlvconverter` doesn't support nonstandard resolution conversion. Use the OpenGL for embedded systems (GLES) engine as the convertor on this platform.
- When using the `qtioverlay` plugin with detection models, frame drops may occur, especially with many detections. Use qtivcomposer for detection-based ML use cases.
- Speed-based video playback isn't supported in this release.
- A drop of 1–2 fps may be observed with the three-stream camera use case.

6 Build and flash instructions

Recommended procedure

To upgrade using Qualcomm prebuild, see Update the software.

Advanced procedure

- To build the QIMP SDK, see Build platform image with QIMP layer.
- To flash the QIMP SDK, see Flash software images.

7 References

7.1 Related documents

Title	Document number
Qualcomm Intelligent Multimedia Software Development Kit (IM SDK) Reference	80-70018-50
Qualcomm Intelligent Multimedia Product (QIMP) SDK Quick Start Guide	80-70018-51
RB3 Gen 2 Quick Start Guide	80-70018-253
Qualcomm Linux Build Guide	80-70018-254

7.2 Acronyms and terms

Acronym	Definition
AI	Artificial intelligence
EIS	Electronic image stabilization
EOS	End of stream
GLES	OpenGL for embedded systems
GOP	Group of pictures
GstD	Gstreamer Demon
HDR	high dynamic range
IM SDK	Qualcomm Intelligent Multimedia SDK
IoT	Internet of Things
ISO	International organization for standardization
LTR	long term reference
ML	Machine learning
MQTT	Message queuing telemetry transport
QIMP SDK	Qualcomm Intelligent Multimedia Product Software Development Kit
QBC	Quad bayer coding
QP	Quantization parameter

Acronym	Definition
ROI	Region of interest
RTSP	Real-time streaming protocol
UBWC	Universal band width compression
V4L2	Video for Linux
VGA	Video graphics array
VFS	Virtual file system

LEGAL INFORMATION

Your access to and use of this material, along with any documents, software, specifications, reference board files, drawings, diagnostics and other information contained herein (collectively this "Material"), is subject to your (including the corporation or other legal entity you represent, collectively "You" or "Your") acceptance of the terms and conditions ("Terms of Use") set forth below. If You do not agree to these Terms of Use, you may not use this Material and shall immediately destroy any copy thereof.

1) Legal Notice.

This Material is being made available to You solely for Your internal use with those products and service offerings of Qualcomm Technologies, Inc. ("**Qualcomm Technologies**"), its affiliates and/or licensors described in this Material, and shall not be used for any other purposes. If this Material is marked as "**Qualcomm Internal Use Only**", no license is granted to You herein, and You must immediately (a) destroy or return this Material to Qualcomm Technologies, and (b) report Your receipt of this Material to qualcomm.support@qti.qualcomm.com. This Material may not be altered, edited, or modified in any way without Qualcomm Technologies' prior written approval, nor may it be used for any machine learning or artificial intelligence development purpose which results, whether directly or indirectly, in the creation or development of an automated device, program, tool, algorithm, process, methodology, product and/or other output. Unauthorized use or disclosure of this Material or the information contained herein is strictly prohibited, and You agree to indemnify Qualcomm Technologies, its affiliates and licensors for any damages or losses suffered by Qualcomm Technologies, its affiliates and/or licensors for any such unauthorized uses or disclosures of this Material, in whole or part.

Qualcomm Technologies, its affiliates and/or licensors retain all rights and ownership in and to this Material. No license to any trademark, patent, copyright, mask work protection right or any other intellectual property right is either granted or implied by this Material or any information disclosed herein, including, but not limited to, any license to make, use, import or sell any product, service or technology offering embodying any of the information in this Material.

THIS MATERIAL IS BEING PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESSED, IMPLIED, STATUTORY OR OTHERWISE. TO THE MAXIMUM EXTENT PERMITTED BY LAW, QUALCOMM TECHNOLOGIES, ITS AFFILIATES AND/OR LICENSORS SPECIFICALLY DISCLAIM ALL WARRANTIES OF TITLE, MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, COMPLETENESS OR ACCURACY, AND ALL WARRANTIES ARISING OUT OF TRADE USAGE OR OUT OF A COURSE OF DEALING OR COURSE OF PERFORMANCE. MOREOVER, NEITHER QUALCOMM TECHNOLOGIES, NOR ANY OF ITS AFFILIATES AND/OR LICENSORS, SHALL BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY EXPENSES, LOSSES, USE, OR ACTIONS HOWSOEVER INCURRED OR UNDERTAKEN BY YOU IN RELIANCE ON THIS MATERIAL.

Certain product kits, tools and other items referenced in this Material may require You to accept additional terms and conditions before accessing or using those items.

Technical data specified in this Material may be subject to U.S. and other applicable export control laws. Transmission contrary to U.S. and any other applicable law is strictly prohibited.

Nothing in this Material is an offer to sell any of the components or devices referenced herein.

This Material is subject to change without further notification.

In the event of a conflict between these Terms of Use and the *Website Terms of Use* on www.qualcomm.com, the *Qualcomm Privacy Policy* referenced on www.qualcomm.com, or other legal statements or notices found on prior pages of the Material, these Terms of Use will control. In the event of a conflict between these Terms of Use and any other agreement (written or click-through, including, without limitation any non-disclosure agreement) executed by You and Qualcomm Technologies or a Qualcomm Technologies affiliate and/or licensor with respect to Your access to and use of this Material, the other agreement will control.

These Terms of Use shall be governed by and construed and enforced in accordance with the laws of the State of California, excluding the U.N. Convention on International Sale of Goods, without regard to conflict of laws principles. Any dispute, claim or controversy arising out of or relating to these Terms of Use, or the breach or validity hereof, shall be adjudicated only by a court of competent jurisdiction in the county of San Diego, State of California, and You hereby consent to the personal jurisdiction of such courts for that purpose.

2) Trademark and Product Attribution Statements.

Qualcomm is a trademark or registered trademark of Qualcomm Incorporated. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the U.S. and/or elsewhere. The Bluetooth® word mark is a registered trademark owned by Bluetooth SIG, Inc. Other product and brand names referenced in this Material may be trademarks or registered trademarks of their respective owners.

Snapdragon and Qualcomm branded products referenced in this Material are products of Qualcomm Technologies, Inc. and/or its subsidiaries. Qualcomm patented technologies are licensed by Qualcomm Incorporated.