Femto Bolt Comparison with Azure Kinect DK

Femto Bolt Comparison with Azure Kinect DK Microsoft announced the end of sale and transfer of the Azure Kinect DK to a partner system in Aug 2023. Orbbec's Femto Bolt is the commercial alternate to Microsoft's Azure Kinect DK that uses Microsoft's industry-proven iToF technology and has identical depth camera operating modes and performance. It is a compact, high-performance device with multi-mode Depth and RGB cameras, internal IMU and USB-C connection for power and data.

This document provides valuable information for developers transitioning to ORBBEC® Femto Bolt from the Microsoft Azure Kinect DK.

Comparison of Product Features with Azure Kinect DK Key Features Femto Bolt Azure Kinect DK Depth identical operating modes and performance

Depth image to Color accuracy Optimized to max<4 pixel; mean<2 pixel max<15.05 pixel; mean<7.8 pixel RGB FOV Slightly reduced to H80° V51° H90° V59° RGB HDR support Supported Not supported Device size More compact, only 64.9mm deep Depth dimension: 125.4mm Multi-camera sync 8-pins for extended functions 2 pin audio cable Power consumption 7.5W max (4.0W avg.) 5.9W max Data interface Dual screw locking USB-C USB-C SDK Orbbec SDK with Orbbec SDK K4A Wrapper

Azure Kinect Sensor SDK For details on Femto Bolt hardware specifications, please refer to Hardware Specifications.

Detailed Comparison with the Azure Kinect DK ● Depth Cameras Femto Bolt integrates Microsoft's 1 megapixel Time-of-Flight (ToF) depth camera that is used in the Azure Kinect DK, and provides identical operating modes and performance as described below:

Mode Resolution FoV FPS Operating Range* Exposure Time NFOV unbinned 640×576 75°x65° 5, 15, 25, 30 0.5 – 3.86 m 12.8 ms NFOV 2×2 binned (SW) 320×288 75°x65° 5, 15, 25, 30 0.5 – 5.46 m 12.8 ms WFOV 2×2 binned 512×512 120°x120° 5, 15, 25, 30 0.25 – 2.88 m 12.8 ms WFOV unbinned 1024×1024 120°x120° 5, 15 0.25 – 2.21 m 20.3 ms Passive IR 1024×1024 N/A 5, 15, 25, 30 N/A 1.6 ms *850nm at 15% to 95% reflectivity, 2.2 µW/cm2/nm, standard deviation of random error ≤ 17 mm, typical system error < 11 mm + 0.1% of distance (no multi-path interference).

Depth may be provided beyond the indicated operating range depending on object reflectivity.

● SDK Compatibility The Femto Bolt uses Orbbec SDK for setup and control. In addition, Orbbec has developed the Orbbec SDK K4A Wrapper to make it easier for applications developed based on Azure Kinect DK to be switched to the Femto Bolt. For detailed instructions, please refer to Orbbec SDK K4A Wrapper on GitHub.

● RGB Camera The Femto Bolt RGB camera provides the same resolution as the Azure Kinect DK (3840 x 2160@30fps), but with a slightly smaller FOV. However, the Femto Bolt RGB camera supports HDR, has a high dynamic range and can provide higher quality RGB images and videos.

Femto Bolt Azure Kinect DK RGB FOV H80° V51° H90° V59° RGB HDR Supported Not supported RGB dynamic range Improved to 81.1DB 45.6DB Comparison of color images:

The table below shows a comparison of the camera field of view (what the sensor "sees") for depth and RGB cameras at a distance of 2m.

Comparison of color performance:

The detailed specification comparison of the color cameras is shown in the table below:

Color Camera Characteristics Femto Bolt Azure Kinect DK Pixel pitch 1.45 um 1.25 um Sensor Size 5.568×3.840mm 5.120×3.132 mm F number 2.7 2.2 Aspect Ratio 16:9 4:3 RGB camera resolution (HxV) and Aspect ratio Frame rate (FPS) Format options FOV (HxV) Frame rate (FPS) Format options FOV (HxV) 3840×2160 16:9 0, 5, 15, 25, 30 MJPEG

80°x51° 0, 5, 15, 30 MJPEG

90°x59° 2560×1440 16:9 0, 5, 15, 25, 30 MJPEG

80°x51° 0, 5, 15, 30 MJPEG

90°x59° 1920×1080 16:9 0, 5, 15, 25, 30 MJPEG

80°x51° 0, 5, 15, 30 MJPEG

90°x59° 1280×720 16:9 0, 5, 15, 25, 30 MJPEG/YUY2/NV12

80°x51° 0, 5, 15, 30 MJPEG/YUY2/NV12

90°x59° 4096×3072 4:3 N/A 0, 5, 15 MJPEG

90°x74.3° 2048×1536 4:3 N/A 0, 5, 15, 30 MJPEG

90°x74.3° 1280×960 4:3 0, 5, 15, 25, 30 MJPEG

65°x51° N/A ● D2C Optimization Improvements of D2C accuracy of the Femto Bolt are shown below.

Comparison of foreground and background boundary alignment:

● SDK Difference Orbbec SDK K4A Wrapper Interface Differences

No. Function Point Orbbec SDK K4A Wrapper Azure Kinect Sensor SDK Impact on Application 1 Recording c++ typedef struct _k4a_record_configuration_t { /** * The timestamp offset of the start of the recording. All recorded timestamps are offset by this value such that * the recording starts at timestamp 0. This value can be used to synchronize timestamps between 2 recording files. / *uint64_t start_timestamp_offset_usec; } k4a_record_configuration_t; c++ typedef struct _k4a_record_configuration_t { /* * The timestamp offset of the start of the recording. All recorded timestamps are offset by this value such that * the recording starts at timestamp 0. This value can be used to synchronize timestamps between 2 recording files. */ uint32_t start_timestamp_offset_usec; } k4a_record_configuration_t; Need to replace and recompile the header file with Orbbec SDK K4a Wrapper Orbbec SDK K4A Wrapper Unimplemented Interfaces (Return Empty Value or Exception State)

No. Azure Kinect Sensor SDK Interface Meaning 1 c++ k4a_result_t k4a_set_allocator(k4a_memory_allocate_cb_t allocate, k4a_memory_destroy_cb_t free) Pass in external user-defined memory manager for SDK internal memory application

2 c++ void k4a_capture_set_temperature_c(k4a_capture_t capture_handle, float temperature_c) Set temperature information for capture

3 c++ float k4a_capture_get_temperature_c(k4a_capture_t capture_handle) Get temperature information for capture

4 c++ void k4a_image_set_exposure_usec(k4a_image_t image_handle, uint64_t exposure_usec) Set exposure value for image

5 c++ void k4a_image_set_white_balance(k4a_image_t image_handle, uint32_t white_balance) Set white balance value for image

6 c++ void k4a_image_set_iso_speed(k4a_image_t image_handle, uint32_t iso_speed) Set ISO speed for image

7 c++ uint64_t k4a_image_get_exposure_usec(k4a_image_t image_handle) Get exposure value for image

8 c++ uint32_t k4a_image_get_white_balance(k4a_image_t image_handle) Get white balance value for image

9 c++ uint32_t k4a_image_get_iso_speed(k4a_image_t image_handle) Read ISO speed for image

10 c++ k4a_result_t k4a_device_get_sync_jack(k4a_device_t device_handle, bool *sync_in_jack_connected, bool *sync_out_jack_connected) Get sync cable connection status for device

● Mechanical Femto Bolt is much more compact with about half the depth of the Azure Kinect DK. The spacing between the M2.5 x 4mm mounting screws on the sides had to be reduced.

The USB-C connector can be locked using two screws to enhance mechanical stability and data transmission.

● Multi-Camera Synchronization Function Femto Bolt uses a 8-pin interface with extended and precise synchronization capability compared to the 2 pin audio cable used by Azure Kinect DK.

Electrical integration Femto Bolt Azure Kinect DK Multi-camera synchronization interface 8-pin interface Sync-out, Sync-in Orbbec's 8-pin design combines the sync in and sync out port with advanced trigger control through the SDK. And the trigger voltage is switchable (1.8V, 3.3V, 5V). The number of secondary devices can be extended using multiple hubs.

Orbbec's Sync Hubs are available in two versions, the Sync Hub Dev and Sync Hub Pro, and are designed to reduce the complexity and cost of a multi-camera network of compatible Orbbec cameras and external sensors.

The Developer edition is effective for rapid prototyping during early-stage development. The Professional edition, designed for commercial use, employs a reliable RJ45 interface for a more stable and longer-range signal connection using standard CAT5 or better cables.

When using Orbbec SDK K4A Wrapper, the Femto Bolt multi-camera synchronization function and operation flow are basically consistent with Azure Kinect DK, but:

Femto Bolt does not support sync cable status detection. The Femto Bolt multi-camera synchronization scheme is asynchronous time assignment, while Azure Kinect DK is synchronous reset to zero, which may cause the timestamps to fail to synchronize when multiple Femto Bolts are currently connected to different hosts and used through Orbbec SDK K4A Wrapper. SDK and application layer differences are described in detail:

Module Function Femto Bolt + Orbbec SDK K4A Wrapper Femto Bolt + Orbbec SDK Azure Kinect DK Multi machine timestamp synchronization

Timestamp synchronization clearing N/A Freely configurable and triggered via the interface Zeroing of the first frame after opening the stream in primary and secondary modes Asynchronous timing (PC timing to each device) Timing is turned on by default when the device is turned on Timing can be performed freely through the interface N/A Synchronization cable Connection Status Detection N/A N/A Sync in jack and sync out jack connection detection

Simultaneous Image Acquisition

Synchronization mode within the camera In standalone mode, depth and RGB are synchronized. In standalone mode, depth and RGB are synchronized. In Free Run mode, depth and RGB are individually configurable (No synchronization, different frame rates configurable)

Standalone mode, depth and RGB synchronization Multi-computer synchronization mode Primary + Secondary Primary + Secondary Primary + Secondary Primary-Secondary Delay Configuration Support for configuring the delay of the secondary device relative to the primary device (via depthDelay) Supports depthDelay, colorDelay, triggerOutDelay Supports configuration of

secondary relative to primary delay Depth and RGB Delay Configuration Support for configuring depth or RGB delays (via depthDelay and ColorDelay) Support depthDelay, colorDelay Supports configuration of depth or RGB delay Primary-Secondary opening sequence Recommend turning on the secondary first Recommend turning on the secondary first Required to turn on the secondary first. For related instructions, please refer to: Synchronize Multiple Devices

● Other Electrical Differences Electrical Integration Femto Bolt Azure Kinect DK DC DC 12V/2A DC plug dimensions:

Inner diameter = 2.1mm; Outer diameter = 5.5mm; Pin diameter = 0.6mm

DC 5.2V/2.5A DC plug dimensions:

Inner diameter = 3.0mm; Outer diameter = 4.5mm; Pin diameter = 0.6mm

Multi-camera sync interface 8-pin interface Sync-out, Sync-in Average power consumption 4.35W 3.9W Max power consumption Max power consumption < 8.8W Max power consumption < 5.9W ● IMU IMU Femto Bolt Azure Kinect DK IMU sampling frequency Default 500Hz, supports 50Hz, 100Hz, 200Hz, 500Hz, 1KHz, 2KHz 1.6 kHz

---