

LVX
Specifications

v1.1.0.0

Approved by Livox 08/20/2019

CONTENTS

1	LVX Format v1.0	3
2	LVX Format Definition.....	3
3	Data Types.....	3
4	Public Header Block.....	4
5	Devices Info Block.....	5
6	Point Data Block.....	7

1 LVX Format v1.1

This document describes the specifications of LVX format v1.0. The LVX file is a point cloud file format developed by Livox Tech, based on the company's LiDAR sensors. This file format allows users to play the point cloud file at a base frequency of 20 Hz. At the same time, users can also acquire point data from a single device from this file for more complex algorithm development.

2 LVX Format Definition

The format contains binary data consisting of public header block, device info block, and point cloud data block.

PUBLIC HEADER BLOCK
PRIVATE HEADER BLOCK
DEVICE INFO BLOCK
POINT DATA BLOCK

All data are in **little-endian** format. The header block consists of file signature, version information, and a magic code. The length of the devices info block is variable, capable of accommodating any number of devices. The point cloud data block has point cloud data organized by package, and these packages are organized by frames in each file.

3 Data Types

The following data types are used in the LVX format.

- char (1 byte)
- unsigned char (1 byte)
- short (2 bytes)
- unsigned short (2 bytes)
- int (4 bytes)

- unsigned int (4 bytes)
- long long (8 bytes)
- unsigned long long (8 bytes)
- float (4 bytes IEEE floating point format)
- double (8 bytes IEEE floating point format)

4 Public Header Block

Item	Format	Size
File Signature ("livox_tech")	char[16]	16 bytes
Version-A	char	1 bytes
Version-B	char	1 bytes
Version-C	char	1 bytes
Version-D	char	1 bytes
Magic Code	unsigned int	4 bytes

File Signature: The file signature must contain "livox_tech" as it is required by the LVX specification. These characters can be checked by Livox Viewer as an initial determination of file type. Note that the first 10 bytes should be "livox_tech", and the last 6 bytes should be zero filled.

Version: Version a is 1, version b is 1, version c is 0, version d is 0.

Magic Code: This field should be a value of 0xAC0EA767. Livox Viewer will not identify a lvx file with an incorrect Magic Code.

5 Private Header Block

Item	Format	Size
Frame Duration	unsigned int	4 bytes
Device Count	unsigned char	1 byte

Frame Duration: The duration of one frame. The unit of duration is millisecond(ms);
 Note: This field is only used to inform the user of the frame duration of the current file.
 In the 1.1.0.0 version of the Lvx file, this field is 50 and cannot be changed.

Device Count: The length of device info block is variable to suit several devices. This field should be a value of the count of devices;

6 Devices Info Block

Item	Format	Size
Device Info 0	struct	59 bytes
.....		
Device Info N	struct	59 bytes

Device Info: This is a field that provides information of each device. This field is defined as:

Item	Format	Size	Description
LiDAR SN Code	char[16]	16 bytes	LiDAR broadcast code
Hub SN Code	char[16]	16 bytes	Hub broadcast code. Note that an empty hub SN means there is no hub connecting this LiDAR.

Device Index	unsigned char	1 byte	Index in device info list
Device Type	unsigned char	1 byte	Device type: 0: LiDAR Hub 1: Mid 40/Mid-100 2: Tele-15 3: Horizon
Extrinsic Enable	unsigned char	1 byte	0: Extrinsic parameters disable, cloud points should be computed without extrinsic parameters; 1: Extrinsic parameters enable, cloud points should be computed with extrinsic parameters;
Roll	float	4 bytes	Extrinsic parameters: Roll Angle, Unit: degree
Pitch	float	4 bytes	Extrinsic parameters: Pitch Angle, Unit: degree
Yaw	float	4 bytes	Extrinsic parameters: Yaw Angle, Unit: degree
X	float	4 bytes	Extrinsic parameters: X Translation, Unit: m
Y	float	4 bytes	Extrinsic parameters: Y Translation, Unit: m
Z	float	4 bytes	Extrinsic parameters: Z Translation, Unit: m

Note that users can use Device Index to extract point cloud data of each device from a LVX file.

7 Point Cloud Data Block

Data from the Point Cloud Data Block are composed of frames, and each frame is composed of packages.

Item	Format	Size
Frame 0	struct	N bytes
Frame 1	struct	N bytes
.....		
Frame N	struct	N bytes

Frame is defined as:

Item	Format	Size
Frame Header	struct	32 bytes
Package 0	struct	depends
Package 1	struct	depends
.....		
Package N	struct	depends

Frame Header is defined as:

Item	Format	Size	Description
Current Offset	long long	8 bytes	Absolute offset of the current frame in this file.
Next Offset	long long	8 bytes	Absolute offset of next frame in this file.
Frame Index	long long	8 bytes	Current Frame Index.

Package is defined as SDK protocol new type normal Data

Item	Format	Size	Description
Device Index	unsigned char	1 byte	Refer to Device Info
Version	unsigned char	1 byte	Package protocol version, 5 for the current version
Slot ID	unsigned char	1 byte	ID of the Slot Connecting LiDAR Device: For more details, refer to Livox SDK Communication Protocol
LiDAR ID	unsigned char	1 byte	LiDAR ID: 1: Mid-100 Left / Mid-40 / Tele-15 / Horizon 2: Mid-100 Middle 3: Mid-100 Right
Reserved	unsigned char	1 byte	
Status Code	unsigned int	4 bytes	LiDAR Status Indicator Information, refer to Livox SDK Communication Protocol for more details
Timestamp Type	unsigned char	1 byte	Timestamp Type, refer to Livox SDK Communication Protocol for more details
Data Type	unsigned char	1 byte	Point Cloud Coordinate Format: 0: Cartesian Coordinate System; Single Return; (Only for MID) 1: Spherical Coordinate System; Single Return; (Only for MID) 2: Cartesian Coordinate System; Single Return; 3: Spherical Coordinate System;

			Single Return; 4: Cartesian Coordinate System; Double Return; 5: Spherical Coordinate System; Double Return; 6: IMU Information
Timestamp	unsigned char[8]	8 bytes	Nanosecond or UTC Format Timestamp, refer to Livox SDK Communication Protocol for more details
Point 0	struct	depends	Point information, depends on Data Type
Point 1	struct	depends	Point information, depends on Data Type
.....			
Point N	struct	depends	Point information, depends on Data Type

Point is defined as following:

Data type 0 (100 points per package)

Item	Format	Size	Description
x	int	4 bytes	X-axis position, unit: mm
y	int	4 bytes	Y-axis position, unit: mm
z	int	4 bytes	Z-axis position, unit: mm
reflectivity	unsigned char	1 byte	Reflectivity

Data type 1 (100 points per package)

Item	Format	Size	Description
depth	int	4 bytes	Unit: mm
theta	unsigned short	2 bytes	Zenith angle, unit: 0.01 degree,

			range [0,18000]
phi	unsigned short	2 bytes	Azimuth angle, unit: 0.01 degree, range [0,36000]
reflectivity	unsigned char	1 byte	Reflectivity

Data type 2 (96points per package)

Item	Format	Size	Description
x	int	4 bytes	X-axis position, unit: mm
y	int	4 bytes	Y-axis position, unit: mm
z	int	4 bytes	Z-axis position, unit: mm
reflectivity	unsigned char	1 byte	Reflectivity
tag	unsigned char	1 byte	For more detail, please refer to SDK Protocol

Data type 3 (96points per package)

Item	Format	Size	Description
depth	int	4 bytes	Unit: mm
theta	unsigned short	2 bytes	Zenith angle, unit: 0.01 degree, range [0,18000]
phi	unsigned short	2 bytes	Azimuth angle, unit: 0.01 degree, range [0,36000]
reflectivity	unsigned char	1 byte	Reflectivity
tag	unsigned char	1 byte	For more detail, please refer to SDK Protocol

Data type 4 (48 points per package)

Item	Format	Size	Description
x1	int	4 bytes	X-axis position, unit: mm
y1	int	4 bytes	Y-axis position, unit: mm

z1	int	4 bytes	Z-axis position, unit: mm
reflectivity1	unsigned char	1 byte	Reflectivity
tag1	unsigned char	1 byte	For more detail, please refer to SDK Protocol
x2	int	4 bytes	X-axis position, unit: mm
y2	int	4 bytes	Y-axis position, unit: mm
z2	int	4 bytes	Z-axis position, unit: mm
reflectivity2	unsigned char	1 byte	Reflectivity
tag2	unsigned char	1 byte	For more detail, please refer to SDK Protocol

Data type 5 (48 points per package)

Item	Format	Size	Description
theta	int	4 bytes	Zenith angle, unit: 0.01 degree, range [0,18000]
phi	int	4 bytes	Azimuth angle, unit: 0.01 degree, range [0,36000]
depth1	int	4 bytes	Unit: mm
reflectivity	unsigned char	1 byte	Reflectivity
tag1	unsigned char	1 byte	For more detail, please refer to SDK Protocol
depth2	int	4 bytes	Unit: mm
reflectivity2	unsigned char	1 byte	Reflectivity
tag2	unsigned char	1 byte	For more detail, please refer to SDK Protocol

Data type 6 (1 point per package)

Item	Format	Size	Description
gyro_x	float	4 bytes	Unit: rad/s

gyro_y	float	4 bytes	Unit: rad/s
gyro_z	float	4 bytes	Unit: rad/s
acc_x	float	4 bytes	Unit: g
acc_y	float	4 bytes	Unit: g
acc_z	float	4 bytes	Unit: g