- APROVIS3D -

Analog PROcessing of bioinspired VIsion Sensors for 3D reconstruction

Document Reference:		
Title: **D0.2.3 Data Management Plan – Y4 update (final)**		
Contractor: **UCA**		
Prepared by: **Jean Martinet**		
Document Type: **Deliverable**		
Version: **2**		Pages: **11**
Classification: **External document**		

Document Track

Version	Date	Remarks and Authors
1.0	07/12/2023	First draft (J. Martinet - UCA)
2.0	20/12/2023	Final version (J. Martinet - UCA)

Authors

	Role / Function	Name	Organisation
Prepared by	Project Coordinator/WP0L	J. Martinet	UCA
Checked by	Quality check	All partners	All
Released by	Project Coordinator/WP0L	J. Martinet	UCA
Approved by	Project Coordinator/WP0L	J. Martinet	UCA

TABLE OF CONTENTS

FIGURES

TABLES

No table

1 Introduction

During the implementation of the APROVIS3D project, the management process elaborates and implements a plan to collect, store, preserve, and disseminate the data.

1.1 Purpose

This update of the document D0.2.3 deliverable of the APROVIS3D project provides an up-to-date the Data Management Plan (DMP) implemented in the project, and specifies everything about project-generated data (collection methodology, documentation, use, format, storage, confidentiality, and dissemination during and after the project).

Data reusability is one of the main outcomes of the project; hence all datasets and benchmarks created in the project will be handled through the DMP.

2 Documentation

2.1 Applicable and Referenced Documents

#	Id	Description	Identifier (Ed Rev)	Date
AD1	FPP	Full Project Proposal	1.0	15.01.2019

2.2 Glossary and Terminology

Acronym	Definition
DMP	Data Management Plan
WP	Work Package
RGB-D	Red-Green-Blue-Depth
ROS	Robot Operating System
CSV	Comma-Separated Value
UAV	Unmanned Aerial Vehicle

3 Data Management Plan

The DMP distinguishes data that can be made publicly available, and subsets of strategic data that need to be protected.

3.1 Original plan

Originally, the consortium planned to mainly produce three kinds of data:

- Two types of visual data that will be collected by the sensors mounted on the drone: RGB-D and stereo event sensors data. These data are in a resolution that will be determined experimentally (e.g., 1280x720 at 24 fps videos for the RGB-D camera and 128x128 to 256x256 event streams for the event cameras). The RGB-D data also includes 3D information such as point clouds and disparity data. Due to possible drone instability, the visual data may need to be pre-processed to be cleaned. Data will be processed to make it usable in the algorithms, by using image stabilization and noise filtering tools. Therefore, data may include raw and pre-processed/cleaned versions. This data will be stored in HDF5 data format.
- Data collected on the drone motion itself. This will contain the UAV location (coordinate), speed, and all onboard sensor values. The data will be collected with a certain rate during flights, which will allow synchronizing visual data and drone motion data. This data will be stored in CSV format and ROS bag format.
- Algorithms and software pieces related to SNN processing of visual data (INT, UL) and UAV autonomous guidance (NTUA).

3.2 Updated plan during Y2 to Y4

The data produced during the project is used for training and validation purposes, which will guarantee experimental reproducibility. Some data can also be reused in other contexts such as SLAM or odometry, such as the video data recorded from UAV flights at NTUA and UCA. All data is currently stored in local platforms and shared with partners via the project private repository.

Most of the experimental data (visual and UAV motion) are collected by NTUA. Some preliminary visual data have also been collected by UCA using a consumer UAV as test data. A dozen flights produced enough raw data to train and validate the code. Each flight lasts around 15 minutes, which produces both visual and motion data of reasonable size, that enables the transfer between partners in a rather easy way. We use both the shared platform and the FileSender service, proposed by RENATER the French IT services provider for higher education and research. Both ways ensure confidentiality in data exchange.

During the first two years of the project, four datasets have been collected:

- One video dataset containing **simulated** data with RGB, events by NTUA (NTUA_SimulatedBeachDataset). This dataset is mainly used for testing deep-learning-based algorithms on RGB frames. However, this dataset generates low-quality event data DVS Gazebo plugin because of the nature of simulated frames.

- One video dataset containing **real world** RGB data (UCA_TestBeachDataset, see a sample in Figure 1), that contains 15 H264 MP4 video files of resolution 1920x1080 and 24FPS, with a size of 1.5 Go. The files feature 3 categories or classes: only_sea, only_ground, and mixed_sea_ground.

- One video dataset containing **real world** RGB data by NTUA (NTUA_SimulatedBeachDataset). The data is separated in only_sea and only_ground recording categories in order to be matched with the corresponding event-based ones. We recorded 4.45Gb of RGB data utilizing a ZED 2 stereo camera. The format of the files is H264 mp4 with a resolution of 672x376 and 24fps.

- One event dataset containing **real world** event data acquired with a DVS by NTUA (NTUA_BeachDataset). This data is also separated in only_sea and only_ground recording categories such as the RGB ones. The files contain 62.7Mb in an aedat2 format. These files will be used for the dataset generation through Tonic (see below) and later for the training of the SNN classification algorithm.

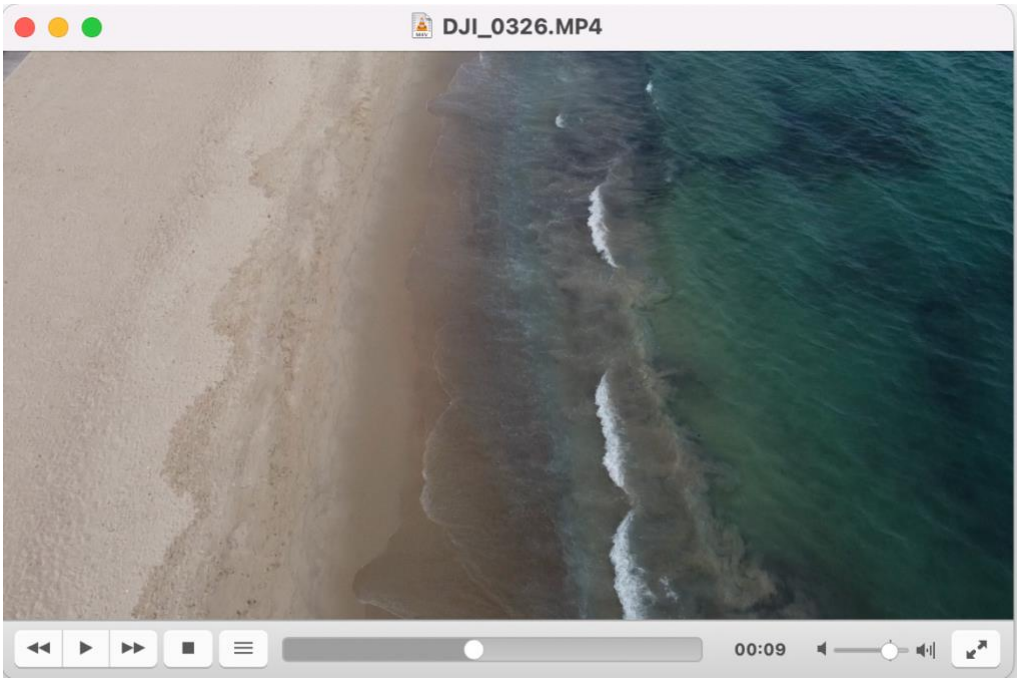

Figure 1: Screen shot of the UCA_TestBeachDataset video dataset

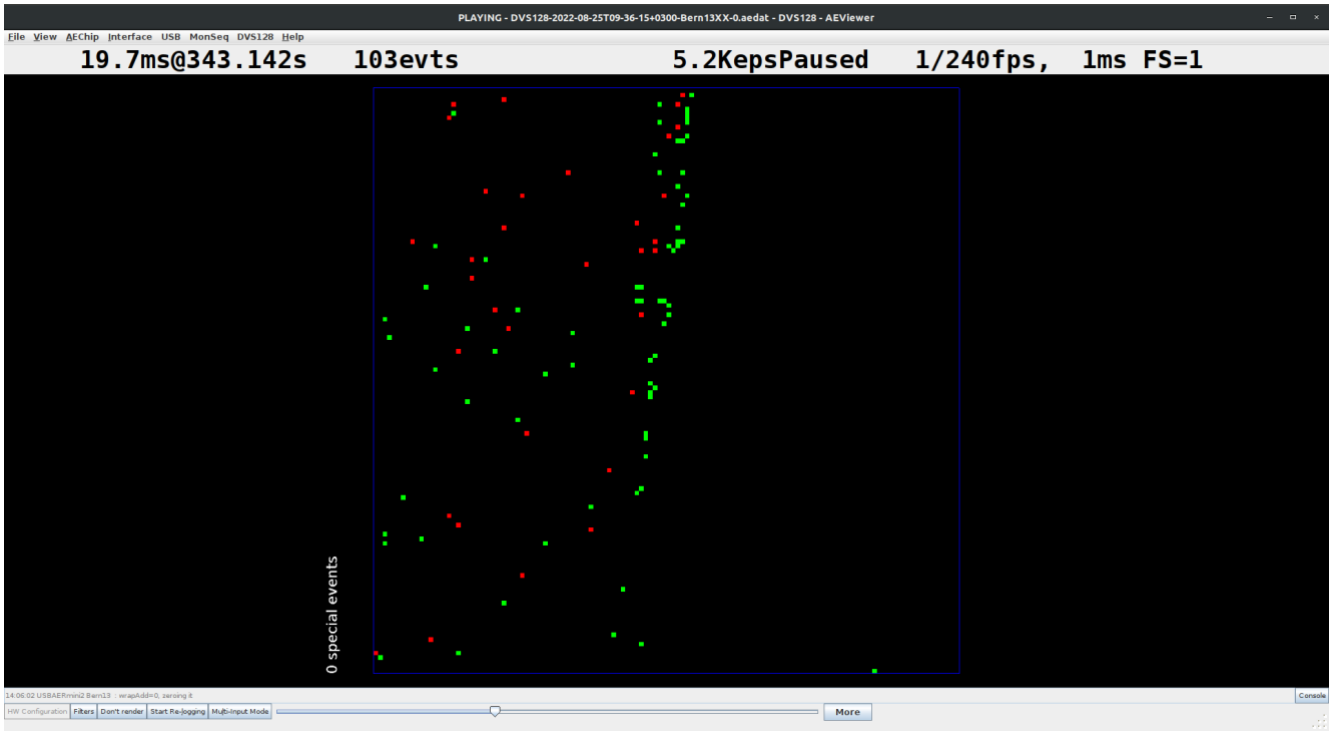

Figure 2: Caption from a Dynamic Vision Sensor (DVS) of a dynamic coastline while flying along its length with an Unmanned Aerial Vehicle (UAV)

Figure 3: Caption from a ZED 2 stereo camera of a dynamic coastline while flying along its length with an Unmanned Aerial Vehicle (UAV)."

Together with the data, we also plan to release the following pieces of code:

- a pipeline to convert it to events, wrap it with Tonic (UCA)
- code for SNN classification with (INT, UCA)
- code for DL classification (NTUA)

All algorithms developed for the project will be shared publicly on the GitHub plateform.

Event-based classification

During these first two years an SNN like event-based classification algorithm was developed by INT and is publicly available at https://github.com/AntoineGrimaldi/hotsline. Tutorials are provided thanks to jupyter notebooks for reproducibility and reusability. These algorithms produce intermediate data to assess their performances, which are stored locally and are not to be shared a priori. They lead to specific results shared through scientific articles.

Video to events

A pipeline to transform RGB MP4 video files into events was implemented by UCA and is publicly available athttps://github.com/amygruel/EvData/tree/master/RGBframes_2_synthetic_events. It uses the Python library vid2e (https://github.com/uzh-rpg/rpg_vid2e) developed by D. Gehrig et al to produce synthetic event data according to different user-defined parameters (frame per second, contrast threshold, etc).

Tonic wrap for synthetic and experimental data

A wrapper was developed by INT and UCA to wrap synthetic data (npy files) and experimental data (aedat2 files) as a Tonic dataset. This wrapper can be found at https://github.com/AntoineGrimaldi/aprovhots/blob/main/dev/dataset_creation_aprovis3d.py ; it splits the input data into patches, which are labeled according to the input sample's name and the user-defined labels. Two tutorials demonstrate how to use this wrapper respectively for synthetic and experimental data:https://github.com/AntoineGrimaldi/aprovhots/blob/main/tutorials/TUTORIAL_2bis_make_a_tonic_dataset_from_RGBframes.py andhttps://github.com/AntoineGrimaldi/aprovhots/blob/main/tutorials/TUTORIAL_2ter_make_a_tonic_dataset_from_AEDAT.py.

Events visualization

An event visualiser has been implemented by UCA to visualise events either accumulated into frames or over time and is publicly available at https://github.com/amygruel/EvVisu/blob/main/visualise_events.py. Relying on the Python library matplotlib, it allows the user to either display or save as gif or png, the event data stored in npy, npz or hdf5 files. Positive and negative events correspond respectively to green and blue dots.

Events reduction

Various methods to reduce the size of event data, either spatially, temporally or structurally, were implemented by UCA and are publicly shared in scientific papers and in the github repository https://github.com/amygruel/EvVisu.

Foveation
Various scripts to detect saliency in and foveate event based data were implemented by UCA and are publicly shared in scientific papers and in the github repository https://github.com/amygruel/FoveationStakes_DVS. The saliency detection consists in a neuromorphic model relying on intrinsic spiking neurons behavior to detect dynamically regions with a high event density.

Data collection and processing during the project does not raise ethics issues but some data will need to be well protected in case we decide to patent or make an economic valorisation of some results. The writing of the Consortium Agreement and of the Data Management Plan has been an opportunity to make sure all partners have a similar approach to balance openness to publications and data while retaining the possibility to patent some of the project results.

All datasets and benchmarks produced during APROVIS3D project are public and advertised through multiple channels (national and European, academic and industrial) namely in publications, where a data availability statement describes how to get the related data.

All publications resulting from this project acknowledge CHIST-ERA, and most of them can be found online from national open repositories (e.g. HAL in France).