PIX2NVS: Parameterized Conversion of Pixel-domain Video Streams to Neuromorphic Vision Streams

Yin Bi and Yiannis Andreopoulos

Dept. of Electronic and Electrical Engineering
University College London (UCL), London, U.K.

* This work is funded by the UCL Overseas Research Scholarship, and the EPSRC project (EP/P02243X/1, IOSIRE Project).

1. Introduction

Neuromorphic vision sensors, a.k.a. dynamic vision sensors or silicon retinas, produce a stream of coordinates and timestamps, labelled as ON or OFF polarity, in an asynchronous manner:

$$E_e = \langle x_e, y_e, t_e, P_e \rangle$$

Dynamic nature makes them popular in many domains such as object tracking, action recognition, or dynamic scene understanding, and any other computer vision fields.

Fig.1. Conventional frames and Neuromorphic vision streams

1. Introduction

Challenge: when working with NVS hardware, there is the lack of annotated datasets.

Limited availability of DVS; Recording data is time & label consuming.

Pixels → Neuromorphic vision streams

Goal: developing a software to generate neuromorphic vision stream datasets from annotated pixel-domain video datasets (e.g. UCF101, YoTube - 8M).

2. Model Description

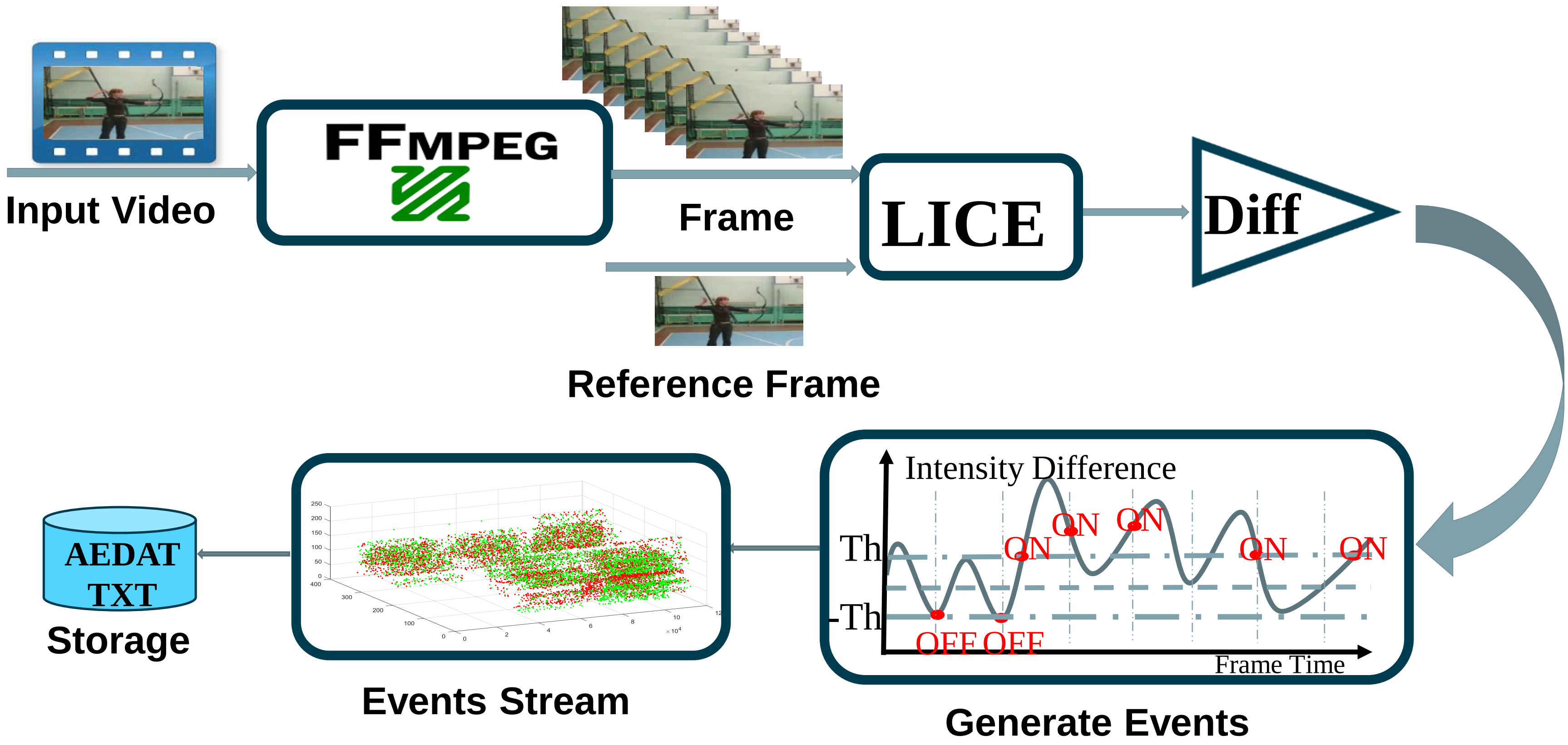


Fig.2. PIX2NVS Framework

2. Model Description

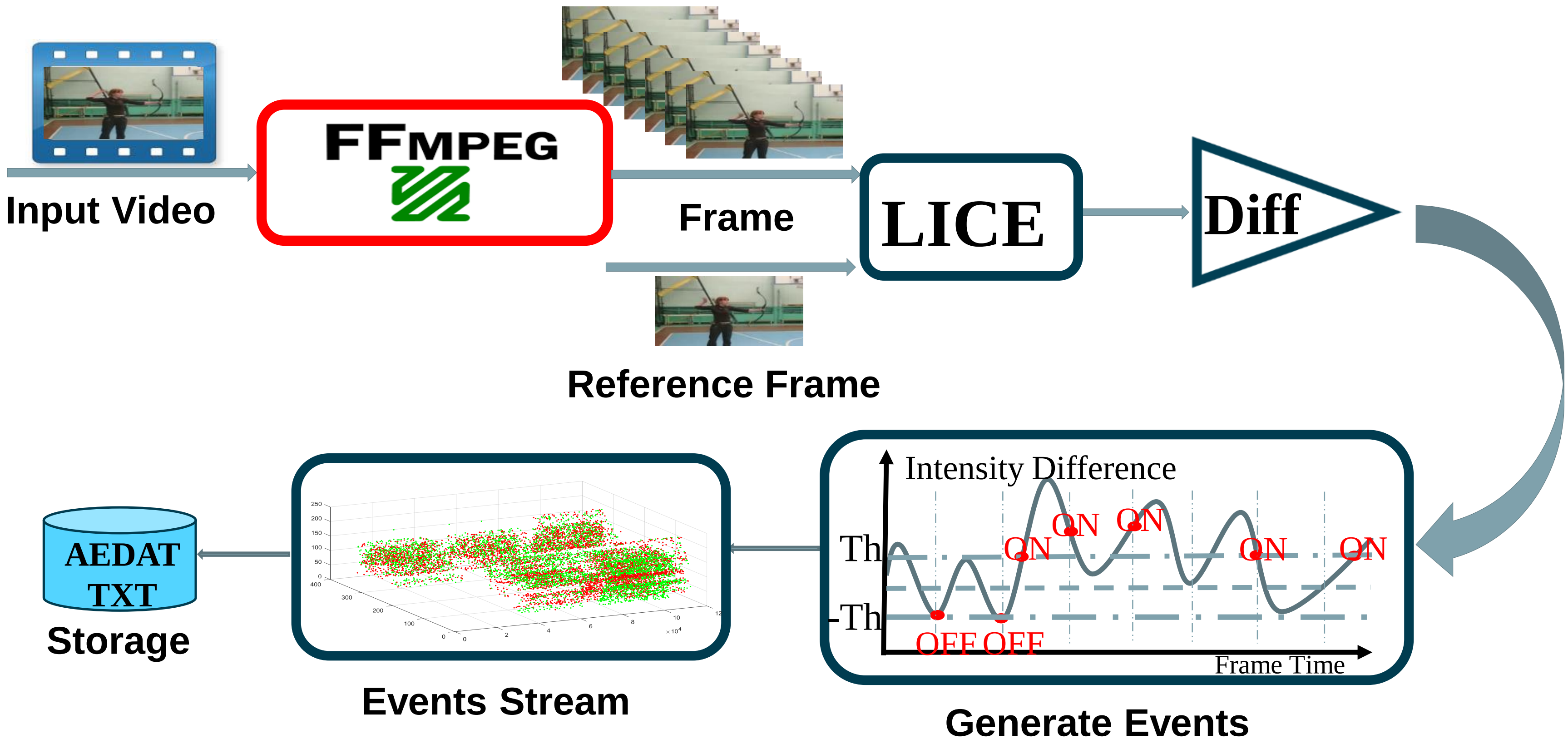


Fig.2. PIX2NVS Framework

2. Model Description

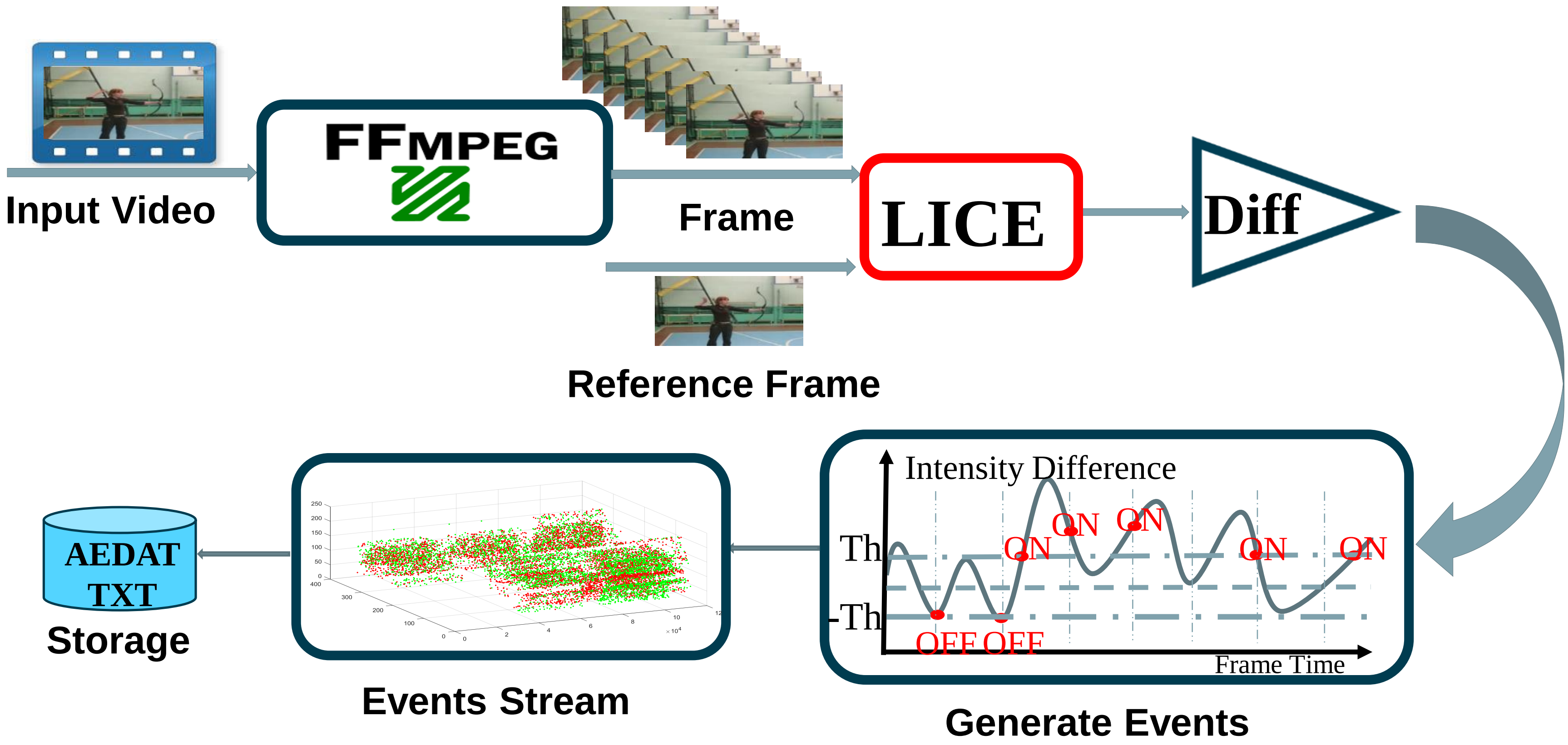


Fig.2. PIX2NVS Framework

2. Model Description

LICE: Converting pixels in position to log-intensity/contrast-enhanced values

1. For RGB values $(r_{i,j}, g_{i,j}, b_{i,j})$ in position (i, j)

If *hue = TRUE:* $y_{i,j} = b_{i,j} / (r_{i,j} + g_{i,j})$

If *hue = FALSE:* $y_{i,j} = 0.299 r_{i,j} + 0.587 g_{i,j} + 0.144 b_{i,j}$

2. *LICE_Mode = {LI,CE}*

If *LICE_Mode = LI:*

$$l_{i,j} = \begin{cases} y_{i,j}, & y_{i,j} \le T_{\log} \\ \ln(y_{i,j}), & y_{i,j} > T_{\log} \end{cases}$$

If *LICE_Mode = CE:*

$$l' = 100 * \sqrt{(y_{i,j} / 255)^{2.2}}$$

$$l_{i,j} = \left(\sum_{p=0}^{1} \left| l'_{i,j} - l'_{i+2p-1,j} \right| + \sum_{p=0}^{1} \left| l'_{i,j} - l'_{i,j+2p-1} \right| \right) / 4$$

2. Model Description

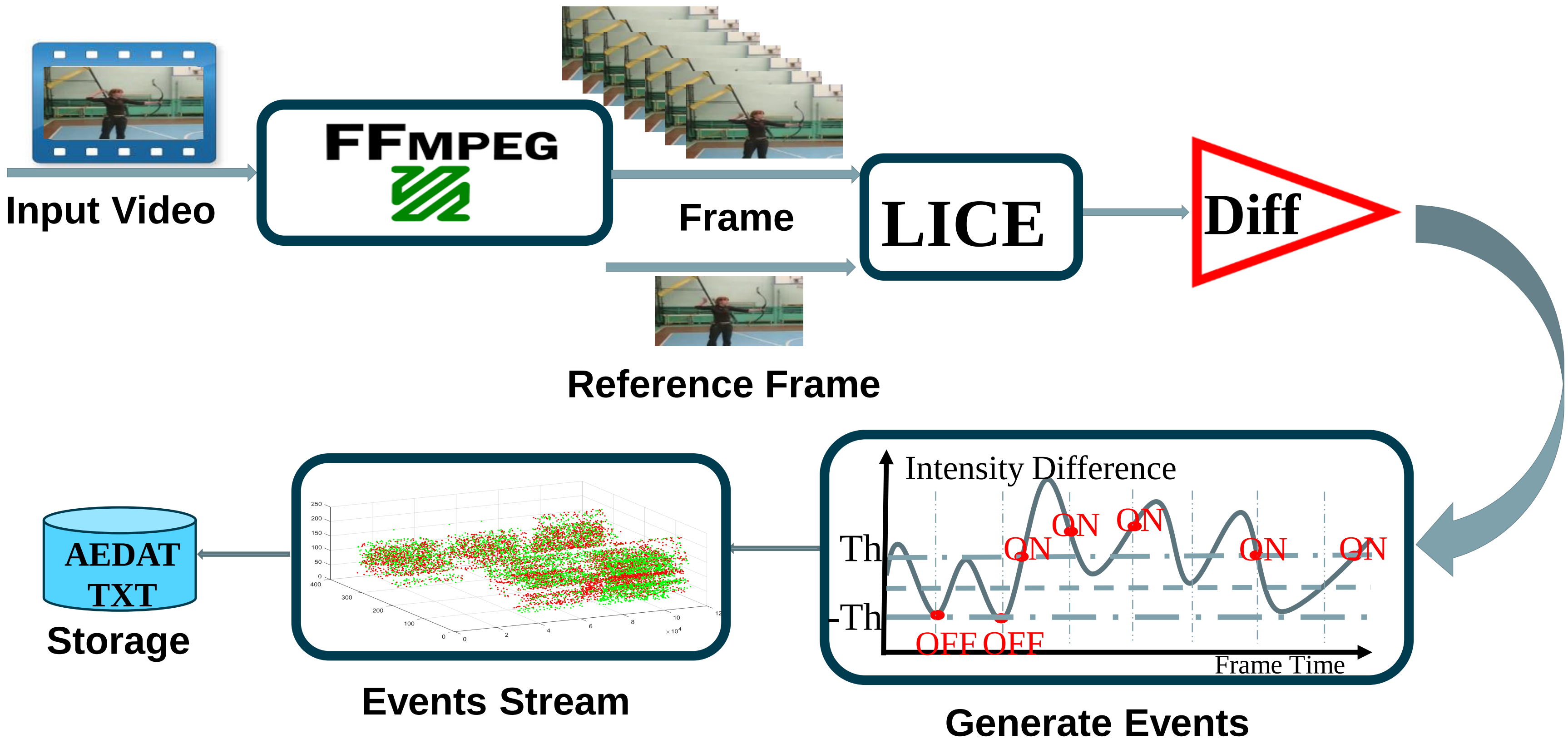


Fig.2. PIX2NVS Framework

2. Model Description

Diff: Establishing difference between frame n and reference frame n-1 (*diff* ={0, *avg*, *min*})

1. If *diff* = 0, i.e. co-located LICE differencing between frames

$$d_{i,j} = l_{i,j}[n] - l_{i,j}[n-1]$$

2. If *diff* = *avg*, i.e. compare to the average of the neighborhood in reference frame

$$d_{i,j} = l_{i,j}[n] - (\sum_{p=0}^{1} l_{i+2p-1,j}[n-1] + \sum_{p=0}^{1} l_{i,j+2p-1}[n-1])/4$$

3. If *diff* = *min*, i.e. compare to the minimum of the neighborhood in reference frame

$$d_{i,j} = l_{i,j}[n] - \min_{p\in 0,1}(l_{i+2p-1,j+2p-1}[n-1])$$

2. Model Description

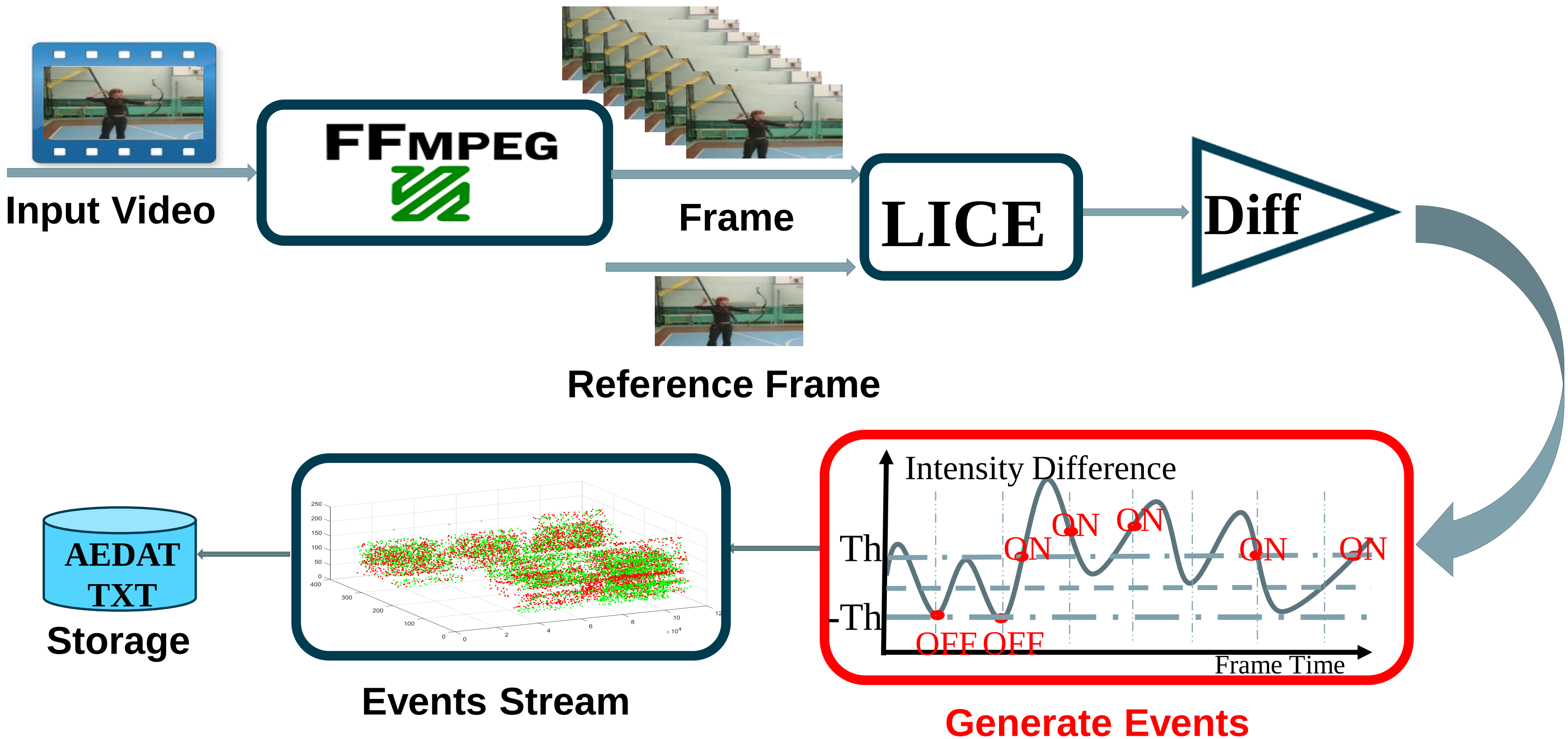


Fig.2. PIX2NVS Framework

2. Model Description

Generate Events: Events are generated if and only if

$$\left|d_{i,j}\right| \geq T_{map}$$

Polarity of the event is

$$P_e = \begin{cases} ON, & \operatorname{sgn}(d_{i,j}) = 1 \\ OFF, & \operatorname{sgn}(d_{i,j}) = -1 \end{cases}$$

Coordinates of the event are

$$(x_e, y_e) = (i, j)$$

2. Model Description

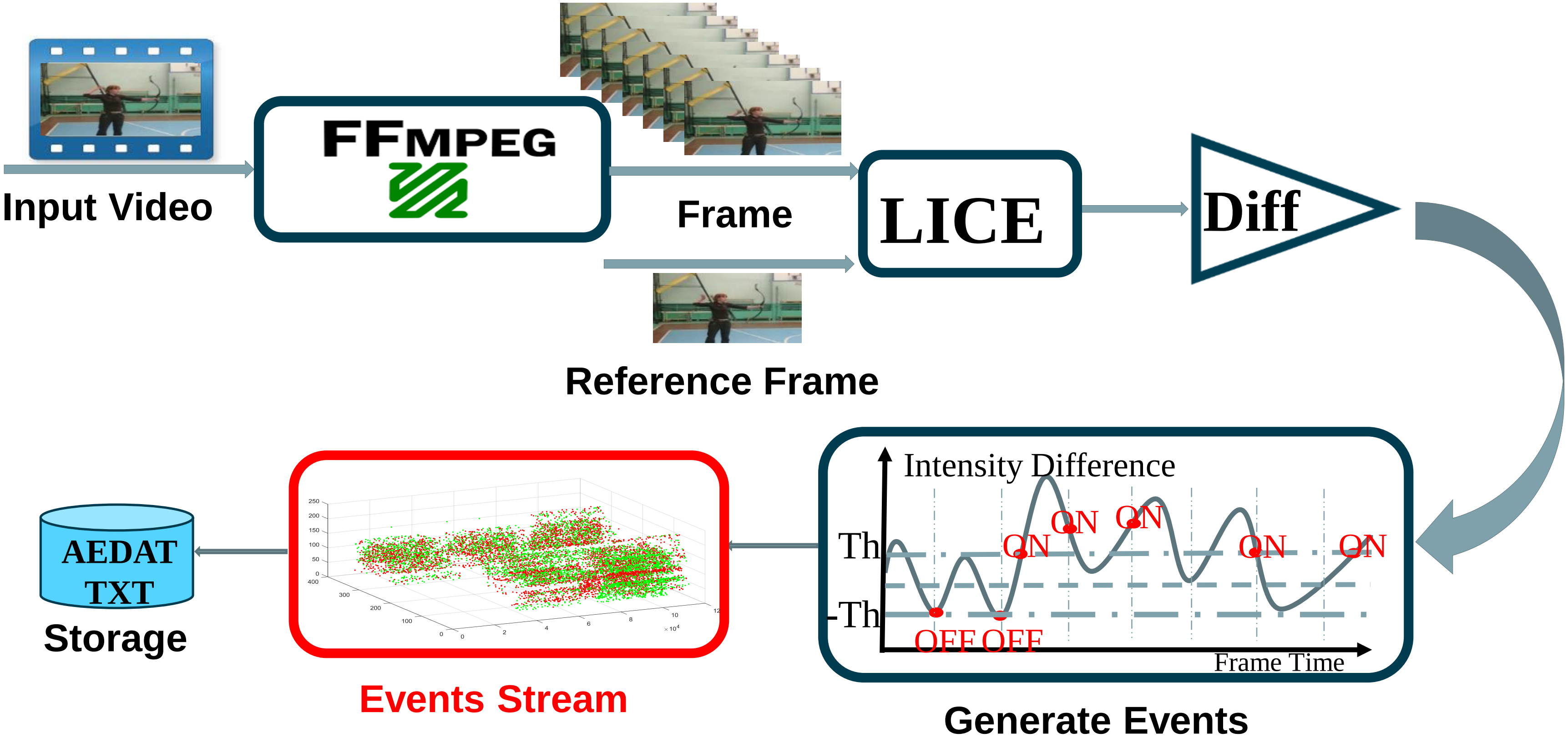


Fig.2. PIX2NVS Framework

2. Model Description

Events Stream: Events are assigned with timestamp (*tstamp={rand, linear, frame}*)

1. If *tstamp* = *rand*, i.e. timestamp is a random number between successive frames

$$t_e = U([n-1, n]) \times fps$$

2. If *tstamp* = *linear*, i.e. timestamp is a linear interpolation number between frames

$$t_e = (n-1+e/e_{tot}[n]) \times fps$$

3. If *tstamp*= *frame*, i.e. timestamp is fixed to frame time

$$t_e = n \times fps$$

2. Model Description

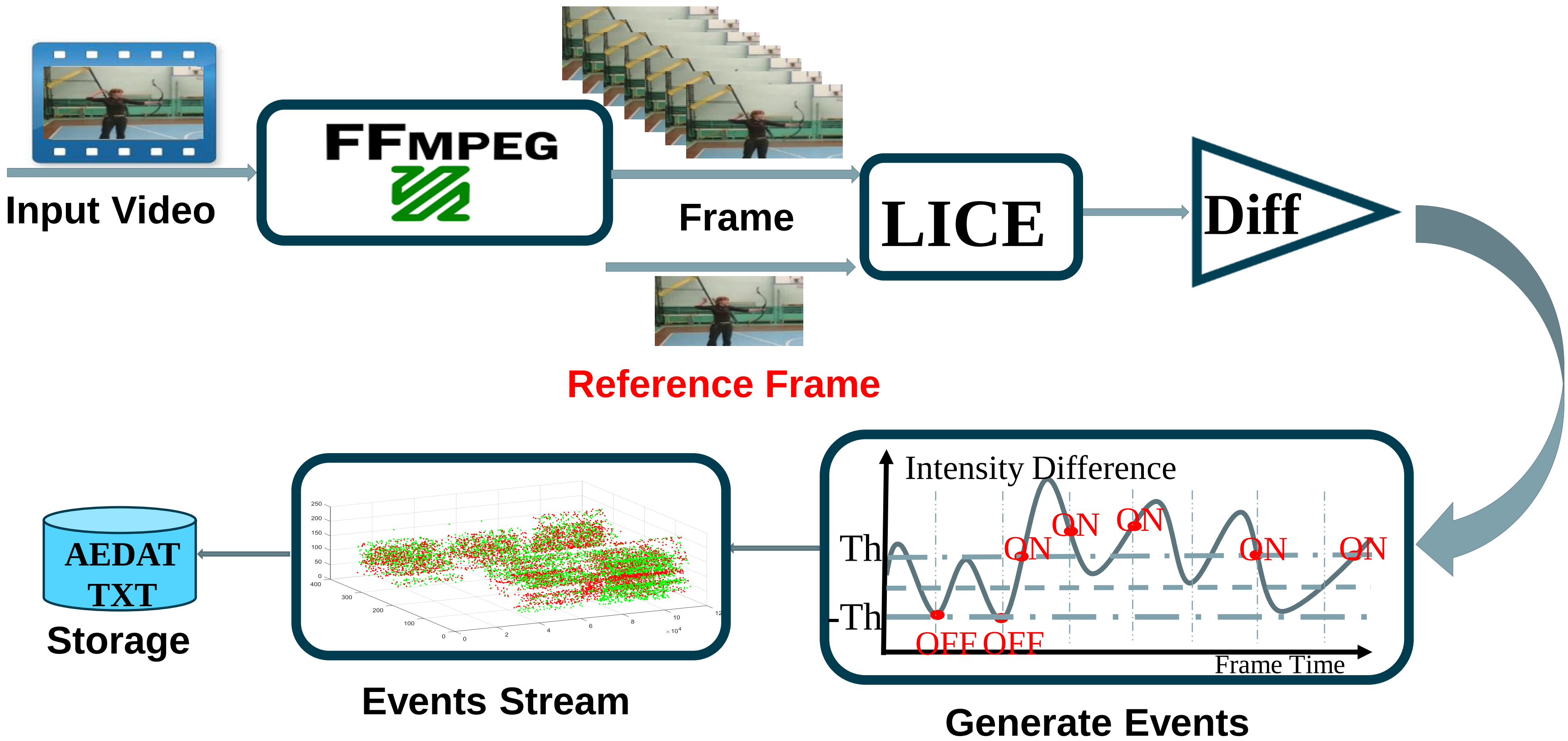


Fig.2. PIX2NVS Framework

2. Model Description

Reference Frame Update: *new={false, true}*

1. If *new* = *true*, i.e. reference frame is new arriving frame *n*

$$l_{i,j}[n] = l_{i,j}[n-1]$$

2. If *new* = *false*, i.e. copy pixel of frame *n* to reference only if this position generates event

$$l_{i,j}[n] = l_{i,j}[n-1], \quad \text{when} \quad (x_e, y_e) = (i, j)$$

2. Model Description

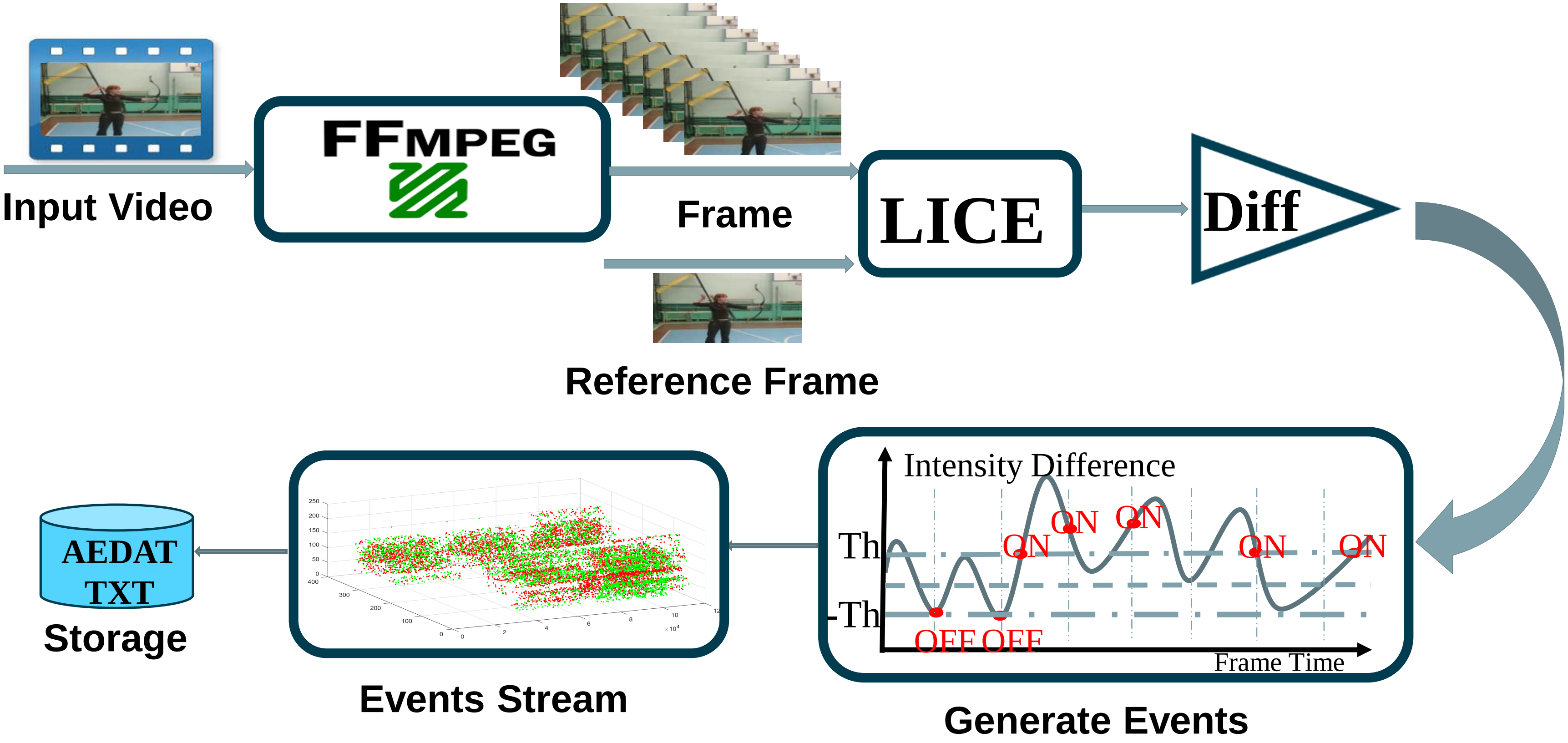


Fig.2. PIX2NVS Framework

3. Display of Generated Events

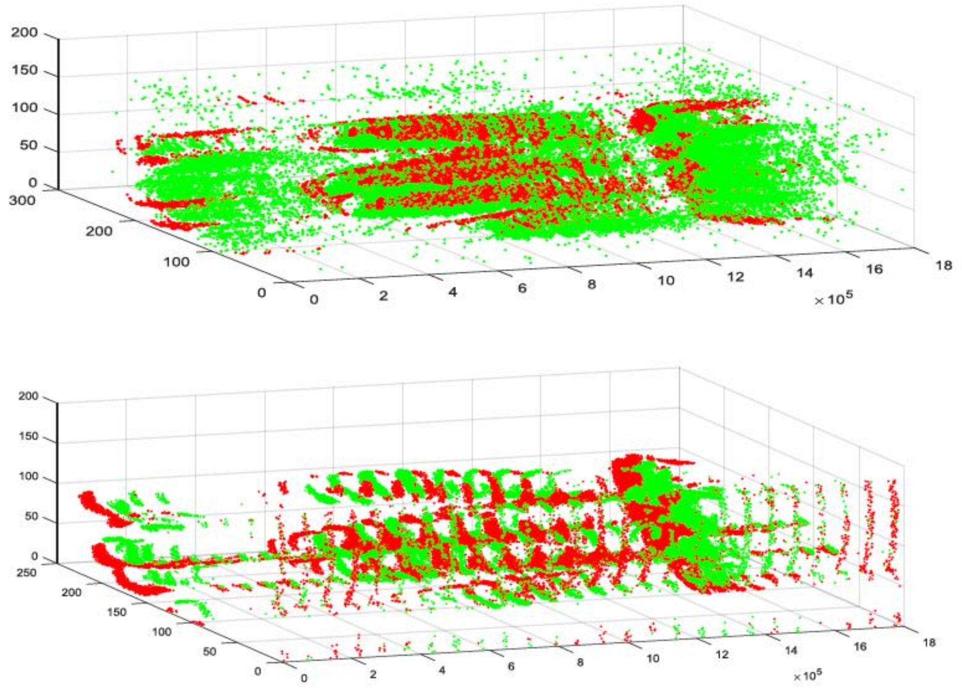


Fig. 3. Experimental NVS events (top) and model-generated ones (bottom). Green/Red points: Trigger ON/OFF.

4. Distance Metrics

To evaluate the performance of PIX2NVS against ground truth NVS data, **Chamfer distance** and **ε–repeatability** are proposed to quantify correspondences.

Chamfer distance: for each model event $E_i^{\text{mod}} =< x_i^{\text{mod}}, y_i^{\text{mod}} >$ (with $E_i^{\text{mod}} \in F_n^{\text{mod}}$), we first search for event $E_j^{\text{exp}} =< x_j^{\text{exp}}, y_j^{\text{exp}} >$ (with $E_j^{\text{exp}} \in F_n^{\text{exp}}$) with the minimium Euclidean distance calculated based on their spatial coordinates

$$j^* = \arg\min_{\forall j} \left\| (x_i^{\text{mod}}, y_i^{\text{mod}}) - (x_j^{\text{exp}} - x_j^{\text{exp}}) \right\|$$

Then Chamfer distance for the $e_{tot}[n]$ model events corresponding to F_n^{mod} is defined as

$$c(n) = \sum_{i=1}^{e_{tot}[n]} \left\| (x_i^{\text{mod}} - y_i^{\text{mod}}) - (x_{j^*}^{\text{exp}} - y_{j^*}^{\text{exp}}) \right\| \Big/ e_{tot}[n]$$

4. Distance Metrics

ε–repeatability: defined as the number of events in F_n^{mod} repeated in $F_n^{\exp}$ within ε distance with respect to the total events.

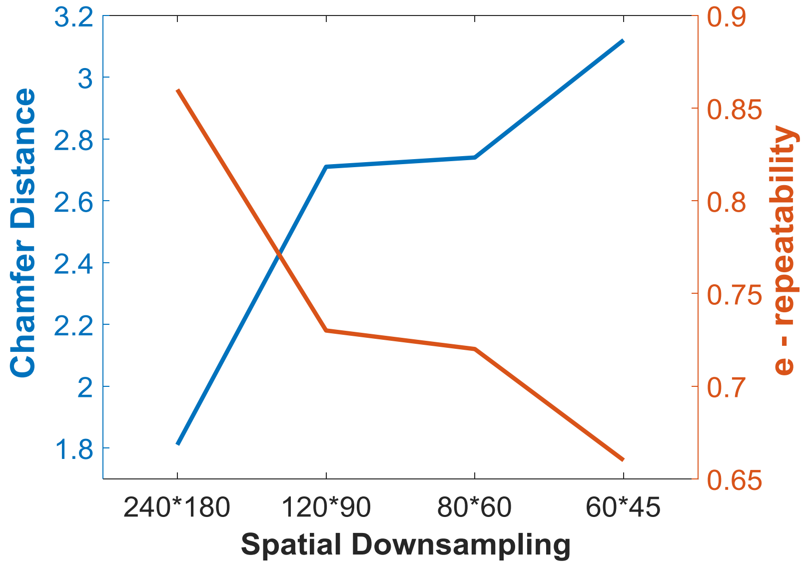
First, we get a new model event set:

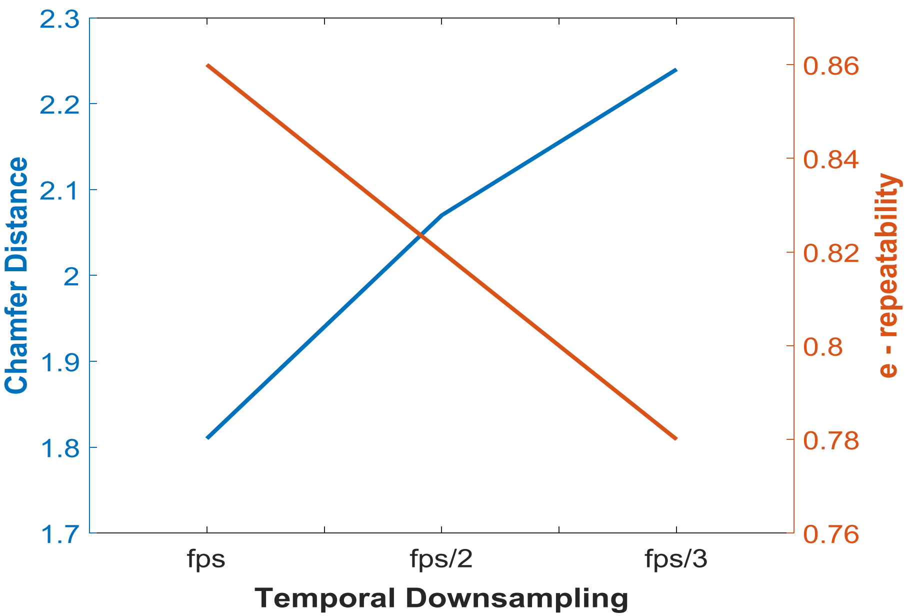
$$E_i^{\mathrm{mod},\varepsilon} = \begin{cases} E_i^{\mathrm{mod}}, & \left\| (x_i^{\mathrm{mod}}, y_i^{\mathrm{mod}}) - (x_j^{\exp} - x_j^{\exp}) \right\| \le \varepsilon \\ \varnothing, & otherwise \end{cases}$$

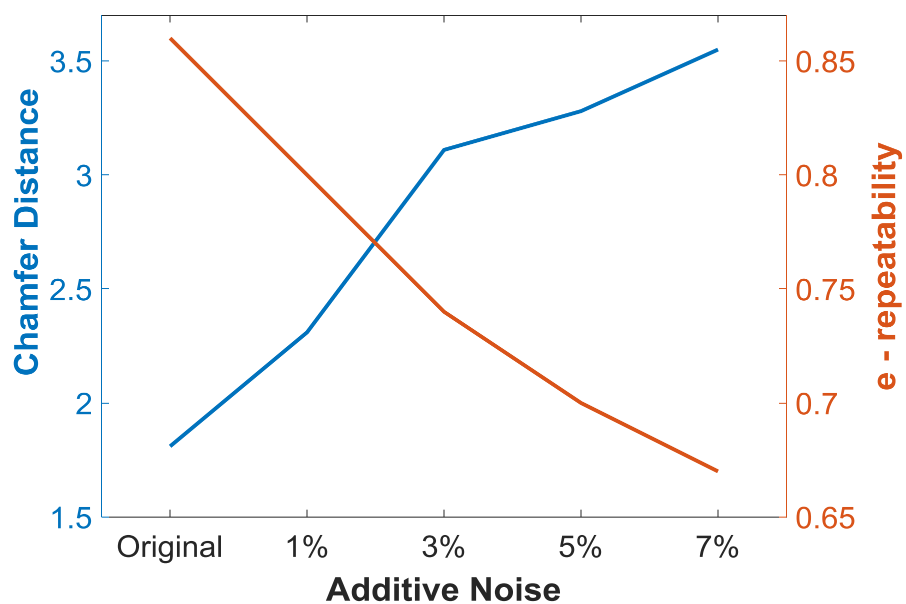
Then the ε-repeatability rate for F_n^{mod} is defined by the normalized l_0 'norm':

$$r^{\varepsilon}[n] = \left| E_i^{\mathrm{mod},\varepsilon} \right| / e_{tot}[n]$$

5. Experimental Validation of Proposed Metrics







6. Initial Validation of Model Options

Comparison	Options	Actual Frame/Dataset	
		Chamfer distance	ε–repeatability
LICE Conversion	*LI*, T_{log}=0	1.81/1.22	0.86/0.90
	LI, T_{log}=20	2.14/1.83	0.82/0.83
	CE	2.54/2.70	0.78/0.78
LICE Checking	*diff = 0*	1.81/1.22	0.86/0.90
	diff = min	2.51/1.48	0.78/0.86
	diff = avg	1.81/1.13	0.86/0.90
LICE Update	*new = true*	1.81/1.22	0.86/0.90
	new = false	1.90/1.24	0.83/0.89

Table 2. Chamfer distance / ε -repeatability (ε = 2.5) w.r.t. different options

7. Conclusion

1. Propose and make available online a parametric tool for software conversion of pixel-domain video frames into neuromorphic vision streams.

   https://github.com/pix2nvs

2. Propose and verify two distance metrics, **Chamfer distance** and **ε-repeatability**, to quantify the accuracy of the model-generated NVS against ground-truth event streams.