

Article

Not peer-reviewed version

Universal Sensor Frontend for Event Inference in Photonic Stochastic Systems

Oleg Angelsky , [Myroslav Strynadko](#) ^{*} , Claudia Zenkova , Roman Zaiats , Xinzheng Zhang , [Jun Zheng](#) ^{*} , [Jingxian Cai](#)

Posted Date: 30 March 2026

doi: 10.20944/preprints202603.2303.v1

Keywords: heterogeneous sensors; stochastic computing; photonic stochastic systems; event-level inference; Bernoulli bitstreams; multisensor fusion; edge frontend; polarization encoding; synchronization; decorrelation



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Universal Sensor Frontend for Event Inference in Photonic Stochastic Systems

Oleg Angelsky ^{1,2}, Myroslav Strynadko ^{1,*}, Claudia Zenkova ^{1,2}, Roman Zaiats ¹, Xinzheng Zhang ³, Jun Zheng ^{2,*} and Jingxian Cai ²

¹ Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine

² Taizhou Institute of Zhejiang University, Taizhou, China

³ Nankai University, Tianjin, China

* Correspondence: m.strinadko@chnu.edu.ua (M.S.); dbzj@netease.com (J.Z.)

Highlights

What are the main findings?

- A universal edge frontend was developed to convert heterogeneous sensor measurements into unified event-oriented probabilities and polarization-compatible Bernoulli optical bitstreams for photonic stochastic processing.
- Reliability-aware weighting, stochastic stream decorrelation, and time-varying synchronization control were identified as key design factors governing event-inference accuracy, robustness, and scaling behavior.

What are the implications of the main findings?

- The proposed architecture enables heterogeneous sensors to interface with photonic stochastic processors through a common event-level representation rather than through raw physical units.
- The reported design rules for weighting, decorrelation, synchronization, and robustness provide a practical basis for scalable multisensor edge inference in future photonic sensing systems.

Abstract

Heterogeneous sensor systems generate measurements in incompatible physical units, which complicates their direct integration with photonic stochastic processors. This study proposes a universal edge frontend that converts heterogeneous sensor channels into unified event-oriented probabilities and then into Bernoulli bitstreams compatible with polarization-encoded optical interfaces. The framework combines sensor-to-probability mapping, weighted event-level fusion, stochastic bitstream generation, and system-level control of correlation and synchronization. Its performance was investigated through reproducible Colab-based modeling using baseline validation, weighting-strategy comparison, static and time-varying decorrelation/synchronization studies, and robustness/scaling analysis. The results show that the stochastic event estimate converges toward the float reference with increasing bitstream length, reliability-aware weighting outperforms equal and tested data-driven weighting in the benchmark, independent stream generation provides the best inference quality, and synchronization mismatch becomes measurable in time-varying fusion. The frontend also demonstrates graceful degradation under channel corruption and favorable scaling under mixed informative, weak, redundant, conflicting, and noisy channel configurations. These findings indicate that heterogeneous sensors can be interfaced with photonic stochastic systems through a common event-level representation and that weighting, decorrelation, synchronization, and robustness must be treated as core frontend design variables.

Keywords: heterogeneous sensors; stochastic computing; photonic stochastic systems; event-level inference; Bernoulli bitstreams; multisensor fusion; edge frontend; polarization encoding; synchronization; decorrelation

1. Introduction

Modern sensing systems increasingly rely on heterogeneous data sources, including environmental, optical, electrical, inertial, and chemical channels, whose outputs differ in physical units, dynamic ranges, sampling rates, and uncertainty profiles. Although such diversity can improve situational awareness and decision quality, it also makes direct joint processing difficult, especially when the target system is not a conventional digital processor but a stochastic or photonic computing platform. In parallel, multi-source information fusion has matured into a broad research field spanning probabilistic, evidence-based, and learning-based methods, while sensor-fusion applications continue to expand in localization, monitoring, and event-detection tasks [1–4]. However, heterogeneous sensor fusion is still most often formulated either in the native physical units of the sensors or after domain-specific feature engineering, which does not naturally provide a standardized interface for photonic stochastic processing.

Stochastic computing offers an appealing conceptual basis for such an interface because probabilities can be represented directly by bitstream statistics, enabling compact arithmetic primitives and tolerance to certain classes of noise and hardware imperfections [5]. Closely related probabilistic hardware concepts, such as p-bits and probabilistic logic, further emphasize the usefulness of tunable stochastic representations for inference-oriented architectures [6]. These ideas are attractive for edge sensing, where the primary requirement is often not exact reconstruction of a raw waveform but reliable estimation of an event, risk, or decision variable from multiple imperfect channels.

At the same time, photonic computing has advanced rapidly from conceptual optical accelerators to high-throughput experimental systems, including photonic convolutional processors and continuous-time tensor-core style architectures [7,8]. Reviews of analog optical computing and intelligent photonics also show that photonic platforms are increasingly being considered for real-time data processing because of their high bandwidth, parallelism, and energy-efficiency potential [9,10]. More recently, edge and in-sensor computing studies have argued that future sensing systems should not merely transmit raw measurements, but should increasingly transform data into computation-ready forms near the sensing node itself [11]. Emerging photonic edge-intelligence demonstrations go even further by showing that multi-modal analog inputs can be fused into optical representations suitable for low-latency processing [12].

Despite these advances, an important architectural gap remains. Existing sensor-fusion studies typically optimize algorithms for specific tasks and sensor combinations, whereas stochastic-computing studies focus on probabilistic representations and arithmetic, and photonic-computing studies focus on optical hardware acceleration. What is still insufficiently developed is a universal edge frontend that converts heterogeneous sensor channels into a common event-oriented stochastic interface that can be consumed by a photonic stochastic processor. In such a frontend, the goal is not to preserve a complete raw signal description or to reconstruct an original waveform, but to map each channel into a normalized contribution to an event of interest, expressed as a probability $p_i \in [0,1]$, and then to encode this contribution into a Bernoulli bitstream compatible with a physical optical interface. This distinction is important: the present work addresses event-level multisensor interfacing, not compact representation of a single signal waveform.

In this paper, we propose a universal edge frontend for heterogeneous sensors in photonic stochastic systems. The frontend maps heterogeneous measurements into a unified event-oriented probability space, generates Bernoulli bitstreams, and defines a polarization-compatible optical abstraction in which stochastic binary states can be carried by orthogonal optical states. On top of this representation, the system performs weighted multisensor fusion to estimate event probability. Beyond the architectural proposal itself, the study systematically investigates the roles of weighting strategy, stochastic stream decorrelation, synchronization, robustness under channel degradation, and scaling with increasing channel count. The computational validation shows that stochastic event

estimates converge to float-reference inference with increasing bitstream length, that reliability-aware weighting is more effective than naive equal weighting in the tested benchmark, that independent bitstream generation outperforms shared-randomness policies, that synchronization mismatch becomes measurable in time-varying fusion, and that the proposed frontend degrades gracefully under missing, noisy, or perturbed channels while benefiting from structured scaling in mixed heterogeneous configurations. These results support the use of a unified event-level stochastic interface as a practical bridge between heterogeneous sensing and future photonic stochastic processors.

All references in the main text to Supplement refer to the corresponding sections of the supplementary package deposited in Figshare [18].

2. Related Work

Heterogeneous sensor fusion has been developed across probabilistic, evidence-based, and learning-based paradigms, with applications ranging from environmental monitoring and pedestrian detection to localization and industrial diagnostics. Recent work in Sensors and related journals shows that combining complementary sensing modalities can improve robustness and decision quality, especially when individual channels are noisy or incomplete. At the same time, these studies usually remain task-specific: they are designed for a fixed sensor set, a fixed inference target, and a conventional digital processing backend rather than for a reusable stochastic interface applicable across heterogeneous sensor classes [13–15].

Stochastic computing provides a different perspective by representing probabilities through bitstream statistics, enabling arithmetic and inference directly in the probability domain. Foundational surveys established the core advantages and limitations of stochastic computing, including compact hardware realization, fault tolerance in certain regimes, and sensitivity to correlation between streams. More recent probabilistic hardware concepts, especially p-bit-based architectures, further reinforced the idea that controlled stochastic representations can support inference-oriented computation rather than exact deterministic reconstruction. However, most such works focus on representation, arithmetic primitives, logic, or hardware behavior, rather than on the problem of interfacing heterogeneous real-world sensors through a unified event-oriented frontend [1,2].

In parallel, photonic computing has advanced rapidly as a platform for high-bandwidth and parallel information processing. More broadly, recent optics and photonics reviews have also emphasized the relevance of coherence-related and structured optical-field concepts for practical optical diagnostics and advanced photonic applications [17]. Experimental demonstrations of photonic accelerators, optical convolutional processors, and tensor-style photonic computing architectures have shown that optical systems can implement computationally meaningful transforms at high throughput. Reviews of analog and intelligent photonics also emphasize the growing relevance of optical computing for artificial intelligence and low-latency signal processing. Still, these studies mainly address the computing substrate itself; they do not usually define how diverse sensor measurements should be normalized, weighted, synchronized, and stochastically encoded before entering a photonic processor [3,4].

A related line of research comes from edge and in-sensor computing. These works argue that modern sensing systems should perform at least part of representation building and inference close to the sensing node, rather than forwarding only raw data to a remote processor. Recent photonic edge-intelligence studies extend this argument by showing that multimodal analog information can be processed in compact optical hardware. Nevertheless, even in these cases, the sensor-to-computation interface is typically application-specific. A general-purpose stochastic frontend for heterogeneous channels, especially one formulated around event-level probabilities and optical bitstream compatibility, remains insufficiently developed [9–11,16].

This gap is particularly important when the target task is not full waveform recovery but event-level inference. In many sensing scenarios, the relevant output is a probability of occurrence, risk,

anomaly, or decision state rather than a faithful reconstruction of each raw signal. From that viewpoint, heterogeneous sensor channels do not need to be fused in their native physical units; instead, they can be mapped into normalized event contributions and then processed in a common stochastic domain. This shift from raw-signal fusion to event-oriented interfacing is central to the present study [13–15].

The present work differs from several adjacent classes of studies. First, unlike conventional multisensor-fusion papers, it focuses on a reusable sensor-to-stochastic interface architecture rather than on a single application-specific fusion rule. Second, unlike stochastic-representation studies centered on a single signal or waveform, it does not treat compact encoding, spectral decomposition, or reconstruction fidelity as the main objective. Third, unlike photonic-computing papers centered on optical hardware acceleration alone, it explicitly addresses the frontend design variables that determine whether heterogeneous sensing data can be meaningfully delivered to a photonic stochastic processor. In this sense, the contribution of the paper lies at the intersection of heterogeneous sensing, stochastic representation, and photonic interfacing, but its primary novelty is architectural: a universal edge frontend for event-level multisensor inference.

3. System Architecture

The proposed system is organized as a universal edge frontend that converts heterogeneous sensor measurements into a common stochastic representation suitable for event-level inference in photonic stochastic systems. The overall architecture of the proposed universal edge stochastic frontend is shown in Figure 1.

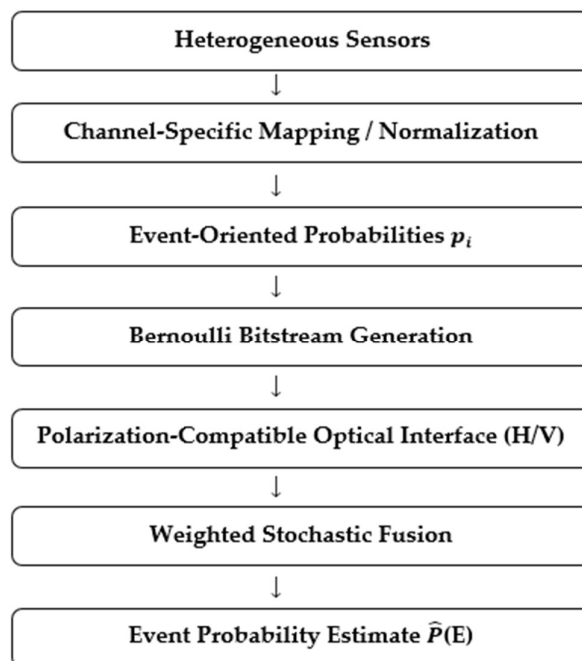


Figure 1. Block diagram of the proposed universal edge stochastic frontend for heterogeneous sensors and its polarization-compatible photonic interface.

Heterogeneous sensor measurements are first converted into channel-specific event-oriented probabilities $p_i \in [0,1]$, then encoded into Bernoulli bitstreams, and finally delivered to a stochastic fusion block through a polarization-compatible optical abstraction in which orthogonal optical states represent stochastic binary states.

The architecture is designed to decouple the physical diversity of primary sensors from the probabilistic format required by the downstream processor. Instead of forwarding raw

measurements in their native units, the frontend transforms each channel into a normalized event-oriented contribution, encodes this contribution into a Bernoulli bitstream, and provides an optical-compatible binary abstraction for subsequent stochastic fusion.

3.1. Heterogeneous Sensor Layer

The input layer may include channels of different physical nature, such as temperature, humidity, wind, rainfall, optical smoke or aerosol sensing, gas concentration, electrical measurements, or other application-specific signals. These channels generally differ in scale, direction of influence, noise characteristics, and temporal behavior. Therefore, they are not directly comparable at the processor interface. In the proposed architecture, each channel is treated first as a source of evidence about a target event rather than as an isolated physical signal. This distinction allows the frontend to operate on heterogeneous sensing systems without requiring a common physical unit across channels.

3.2. Event-Oriented Mapping

For each channel x_i , the frontend applies a channel-specific mapping function that produces a normalized event-related probability-like quantity $p_i \in [0, 1]$:

$$p_i = g_i(x_i), \quad p_i \in [0, 1]. \quad (1)$$

Here, $g_i(\cdot)$ is not merely a numerical normalization operator; it expresses the contribution of channel i to the event of interest. Depending on the sensor semantics, the mapping may be direct or inverse. For a direct channel, larger measured values correspond to larger event contribution. For an inverse channel, larger measured values correspond to smaller event contribution. This formulation enables physically different inputs to be represented in a unified event-oriented space without forcing them into artificial equivalence in their native units.

Figure 2 illustrates how heterogeneous sensor inputs are transformed into event-oriented probabilities and then into Bernoulli bitstreams compatible with the optical interface.

Heterogeneous Sensor Inputs	Event-Oriented Probabilities	Bernoulli Bitstreams
Temperature ($^{\circ}\text{C}$), direct effect	$p_{temp} = 0.72$	$S_{temp}: 10110110$
Humidity (%), inverse effect	$p_{hum} = 0.34$	$S_{hum}: 01000101$
Wind (m/s), direct effect	$p_{wind} = 0.58$	$S_{wind}: 11010100$
Rainfall (mm/h), inverse effect	$p_{rain} = 0.27$	$S_{rain}: 00100101$
Smoke optical, direct effect	$p_{smoke} = 0.81$	$S_{smoke}: 11101011$

Figure 2. Transformation of heterogeneous sensor inputs into a unified stochastic event representation.

Physically different sensor channels with direct or inverse event influence are mapped into normalized event-oriented probabilities and then converted into Bernoulli bitstreams, enabling multisensor fusion in a common stochastic domain.

In practice, $g_i(\cdot)$ may be implemented by linear normalization, thresholded mapping, bounded nonlinear transformation, or a feature-based local preprocessing block. Importantly, such feature extraction is treated here as an auxiliary frontend mechanism, not as the main contribution of the work.

3.3. Stochastic Edge Representation

After event-oriented mapping, each channel probability p_i is converted into a Bernoulli bitstream:

$$S_i \sim \text{Bernoulli}(p_i). \quad (2)$$

In the static formulation, the bitstream encodes the average event contribution of the channel over an observation window. In the time-varying formulation, the frontend produces slot-wise stochastic representations:

$$S_i(t) \sim \text{Bernoulli}(p_i(t)), \quad (3)$$

where t denotes the time slot. This second formulation is especially important when the target event evolves in time and when synchronization between channels must be explicitly controlled.

The use of Bernoulli streams serves two roles. First, it creates a common stochastic protocol for heterogeneous sensor channels. Second, it makes the frontend compatible with stochastic computing and photonic binary-state implementations, where information is naturally processed in probabilistic bitstream form.

3.4. Polarization-Compatible Optical Interface

To bridge the edge frontend and a photonic stochastic processor, the proposed architecture uses a polarization-compatible optical abstraction in which the two logical stochastic states are represented by orthogonal optical states. In the simplest formulation, the Bernoulli states are associated with orthogonal polarization components: $H \rightarrow 1, V \rightarrow 0$.

This representation is not introduced as a complete hardware realization in the present paper, but as a physically meaningful interface layer between edge-generated stochastic bitstreams and downstream photonic processing blocks, which is consistent with the broader trend toward photonic architectures for high-throughput and parallel information processing [3,4]. Such an abstraction is useful because it provides a direct correspondence between stochastic binary coding and optical state encoding, which is compatible with polarization-based photonic logic and measurement schemes.

3.5. Event-Level Fusion Block

The downstream processor operates not on raw sensor values, but on the normalized stochastic contributions of all channels. In the float reference model, multisensor fusion is performed by a weighted aggregation:

$$z = \sum_{i=1}^M w_i p_i, \quad (4)$$

followed by a nonlinear decision mapping:

$$P(E) = \sigma(\alpha(z - \theta)), \quad (5)$$

where w_i are channel weights, θ is the decision threshold, α is a slope parameter, and

$\sigma(\cdot)$ is the sigmoid function. In the stochastic domain, the same logic is applied to empirical bitstream-derived estimates \hat{p}_i or $\hat{p}_i(t)$, producing an estimated event probability $\hat{P}(E)$ or $\hat{P}(E,t)$.

This architecture therefore treats event inference as a probabilistic fusion problem in a common stochastic representation space. The frontend is universal in the sense that heterogeneous sensing modalities are converted into the same interface format before fusion, while the application-specific semantics are captured through the mapping rules and weights.

3.6. Core Design Variables

A key feature of the proposed architecture is that inference quality depends not only on the mapping $x_i \rightarrow p_i$, but also on several system-level design variables:

- **Weight assignment.** The influence of each sensor channel is controlled by w_i , which may be assigned equally, manually, reliability-aware, or by data-driven calibration.

- **Bitstream precision.** The length of the Bernoulli stream determines the stochastic estimation accuracy and convergence to the float reference.
- **Correlation policy.** Independent, grouped, or shared-randomness stream generation changes the effective inter-channel correlation and therefore the fusion quality.
- **Synchronization mode.** In time-varying inference, perfect alignment, fixed lag, or random jitter affect the temporal consistency of multisensor fusion.
- **Robustness conditions.** Noise, channel dropout, missing slots, and weight perturbations influence both average inference error and temporal event tracking quality.

These variables are not secondary implementation details; they are integral parts of the frontend architecture. Accordingly, the remainder of the paper evaluates the proposed system not only as a conceptual interface, but also as a design space with measurable effects on inference accuracy, robustness, and scaling.

3.6. Architectural Scope of the Present Study

The present work focuses on the architectural and computational validation of the frontend. It demonstrates how heterogeneous sensor channels can be transformed into a unified stochastic event-level interface and how this interface behaves under different fusion, synchronization, decorrelation, robustness, and scaling conditions. Detailed reproducible studies supporting these architectural claims are provided in the Supplementary Material: baseline frontend validation in Supplement S1, weighting analysis in Supplement S2, decorrelation analysis in Supplement S3, time-varying synchronization analysis in Supplement S3b, and robustness/scaling analysis in Supplement S4.

If needed, future hardware-oriented extensions may instantiate the proposed interface using specific polarization-based photonic modules, modulators, detectors, and synchronization circuitry. However, the central contribution of the present paper is the universal sensor-to-stochastic frontend architecture itself.

4. Methods

This section formalizes the proposed universal edge frontend and the computational framework used for its validation. The method is organized around five stages: event-oriented channel mapping, float-reference fusion, stochastic bitstream generation, correlation/synchronization control, and robustness/scaling evaluation. The formulation is intentionally architecture-oriented: its purpose is not to reconstruct original waveforms, but to provide a common event-level stochastic interface for heterogeneous sensors.

4.1. Event-Oriented Sensor Mapping

Let x_i denote the measurement of sensor channel i , where $i = 1, \dots, M$. Because channels may differ in units, ranges, and semantics, each channel is converted into a normalized event-oriented quantity (1). For direct channels, larger values of x_i correspond to larger event contribution. For inverse channels, larger values correspond to smaller event contribution. In the simplest bounded linear form,

$$\tilde{x}_i = \frac{x_i - x_{i,min}}{x_{i,max}}, \quad (6)$$

with clipping to $[0,1]$, and

$$\tilde{x}_i, p_i = \begin{cases} \tilde{x}_i, & \text{for direct channels,} \\ 1 - \tilde{x}_i, & \text{for inverse channels} \end{cases} \quad (7)$$

This mapping is interpreted as an event-level transformation, not as a claim that heterogeneous physical quantities become physically equivalent after normalization. Rather, the frontend converts each channel into a common probabilistic contribution format suitable for multisensor fusion.

4.2. Event-Oriented Sensor Mapping

The deterministic reference event score is defined as a weighted combination of the mapped channel probabilities (4), where $w_i \geq 0$ are normalized channel weights satisfying

$$\sum_{i=1}^M w_i = 1. \quad (8)$$

The corresponding float event probability is then computed using a sigmoid decision model (5), where $\sigma(u) = 1/(1+e^{-u})$.

This float model serves as the reference against which stochastic inference is compared. It is not introduced as the only possible fusion law, but as a transparent and reproducible event-level benchmark.

4.3. Bernoulli Bitstream Representation

Each normalized channel contribution p_i is converted into a Bernoulli bitstream (2). For a bitstream of length N , the empirical channel estimate is

$$\hat{p}_i = \frac{1}{N} \sum_{k=1}^N S_{i,k} \quad (9)$$

The stochastic event estimate in the static setting is then

$$\hat{P}(E) = \sigma \left(\alpha \left(\sum_{i=1}^M w_i \hat{p}_i - \theta \right) \right). \quad (10)$$

This formulation allows direct evaluation of the convergence of stochastic inference toward the float reference as the bitstream length increases, consistent with the standard stochastic-computing interpretation of probability through bitstream statistics [1].

4.4. Time-Varying Slot-Wise Formulation

To study synchronization effects, a time-varying formulation is introduced. In this case, each channel is represented by a probability trajectory $p_i(t)$, where $t = 1, \dots, T$ denotes the time slot. The slot-wise Bernoulli representation is (3). For each slot, a local bitstream of length N_{bits} is generated, yielding

$$\hat{p}_i(t) = \frac{1}{N_{\text{bits}}} \sum_{k=1}^{N_{\text{bits}}} S_{i,k}(t). \quad (11)$$

The time-varying stochastic event estimate becomes

$$\hat{P}(E, t) = \sigma \left(\alpha \left(\sum_{i=1}^M w_i \hat{p}_i(t) - \theta \right) \right). \quad (12)$$

This formulation is crucial because synchronization mismatch is largely invisible if inference is based only on full-stream averages. In the slot-wise model, temporal misalignment between channels can directly distort event tracking.

4.5. Weight Assignment Strategies

Four weighting strategies were examined in the computational benchmark. First, **equal weighting** assigns $w_i = 1/M$, treating all channels identically. Second, **expert weighting** uses manually chosen weights derived from assumed channel relevance. Third, **reliability-aware weighting** combines expert importance with a channel reliability factor r_i $w_i \propto \alpha r_i$, followed by normalization to unit sum.

Fourth, data-driven weighting estimates weights from synthetic event labels using a logistic-regression-based calibration procedure. In the present study, this serves as one tested calibration variant rather than as the central method of the paper.

In the present study, this serves as one tested calibration variant rather than as the central method of the paper, whereas the broader relevance of probabilistic hardware-oriented inference is consistent with prior work on tunable stochastic and p-bit-based architectures [2].

These strategies are compared in terms of inference accuracy and threshold consistency.

4.6. Correlation Policies for Stochastic Streams

Because stochastic fusion is sensitive to inter-stream correlation, three stream-generation policies were considered [1].

In the **independent** policy, each channel uses an independent source of randomness.

In the **shared** policy, all channels use a common random sequence, creating strong inter-channel correlation.

In the **grouped** policy, subsets of channels share random sequences within predefined groups, while the remaining channels remain independent. This produces partial correlation and intermediate behavior between the fully independent and fully shared cases.

These policies are used to evaluate how correlation structure affects multisensor event inference.

4.7. Synchronization Policies

Two synchronization formulations were studied.

In the static formulation, synchronization mismatch was introduced as circular shifts within generated bitstreams. This provided a preliminary check, but such shifts do not substantially alter empirical means and therefore only weakly affect full-stream inference.

In the time-varying formulation, synchronization mismatch was applied across the time-slot axis, making it directly relevant for slot-wise event tracking. The following modes were considered:

perfect_sync: no temporal shift,

fixed_lag_1, fixed_lag_4, fixed_lag_8: selected channels shifted by a fixed number of slots,

random_jitter_2, random_jitter_6: channel-wise random shifts within bounded lag intervals.

This second formulation allows synchronization errors to be quantified as a true system-level factor.

4.8. Robustness Scenarios

Robustness was evaluated using four classes of perturbations.

First, **additive channel noise** was injected into selected channels at increasing noise levels.

Second, **single-channel dropout** was implemented by replacing one channel at a time with a neutral value, thereby estimating channel criticality.

Third, **missing-slot corruption** was introduced by replacing a fraction of time slots in selected channels with neutral values.

Fourth, **weight perturbation** was studied using both global random perturbation and targeted modification of dominant or weak channels.

Together, these scenarios test whether the frontend exhibits graceful degradation under realistic forms of degradation or misconfiguration.

4.9. Scaling Configurations

To study scaling, the frontend was evaluated with systems of increasing size: 3, 5, 8, 12, 16 channels.

The added channels were not restricted to uniformly informative channels. Instead, mixed sets were constructed that included informative, weak, redundant, conflicting, and noisy channels. This

makes the scaling study more representative of practical heterogeneous sensor systems, where additional channels are not always equally beneficial.

4.10. Time-Varying Benchmark Construction

For the time-varying studies, synthetic channel probability trajectories were generated using a combination of: baseline channel levels, slow temporal modulation, random fluctuations, localized event-related pulses.

This yielded structured multisensor trajectories in which the event probability evolves over time rather than remaining static. The benchmark was designed not as a domain-specific physical simulator, but as a reproducible architecture-level testbed for studying frontend behavior under controlled heterogeneous conditions.

4.11. Evaluation Metrics

The following metrics were used throughout the study.

The mean absolute error:

$$MAE = \frac{1}{K} \sum_{j=1}^K |y_j - \hat{y}_j|, \quad (13)$$

and the root mean square error:

$$RMSE = \sqrt{\frac{1}{K} \sum_{j=1}^K (y_j - \hat{y}_j)^2}, \quad (14)$$

where y_j and \hat{y}_j denote reference and estimated values.

Bias was defined as

$$Bias = \frac{1}{K} \sum_{j=1}^K (y_j - \hat{y}_j), \quad (13)$$

For event decisions, threshold agreement was computed as the fraction of samples or time slots for which the binarized reference and estimated event decisions matched.

For time-varying inference, temporal consistency was additionally quantified by the **peak alignment error**, defined as the mean absolute difference between the slot index of the reference event peak and the slot index of the estimated event peak.

These metrics were interpreted jointly, because average amplitude error alone does not always capture degradation in temporal localization or threshold stability.

4.12. Reproducibility and Supplementary Implementation

All computational studies were implemented in reproducible Colab notebooks and organized into five supplementary modules: baseline frontend validation (Supplement S1), weighting strategies (Supplement S2), decorrelation study (Supplement S3), time-varying synchronization study (Supplement S3b), and robustness/scaling study (Supplement S4). These supplementary modules provide implementation details, plots, tables, and reproducibility artifacts supporting the main results of the paper.

5. Experimental Design

The experimental design was constructed to validate the proposed frontend at progressively higher levels of architectural complexity. Rather than relying on a single benchmark, the study was organized into a sequence of computational stages that examined baseline interface validity, weight assignment, stochastic correlation, synchronization, robustness, and scaling. This staged design

allowed individual system factors to be isolated and then integrated into a broader evaluation of multisensor event inference.

5.1. Baseline Five-Channel Heterogeneous Benchmark

The core benchmark used throughout the study was a five-channel heterogeneous sensor set composed of: temperature, humidity, wind, rainfall, smoke_optical.

These channels were chosen not to define a single application-specific system, but to represent a realistic heterogeneous mix containing direct and inverse event contributions, different temporal behaviors, and different levels of inferred relevance. In all baseline experiments, these channels were mapped into event-oriented probabilities $p_i \in [0, 1]$, encoded into Bernoulli bitstreams, and fused into an estimated event probability.

The baseline benchmark therefore served as the reference architecture for evaluating whether heterogeneous raw measurements can be transformed into a unified stochastic interface for event-level inference.

5.2. Baseline Validation Stage

The first experimental stage was designed to validate the frontend itself. In this stage, heterogeneous raw channels were generated synthetically, mapped into event-oriented probabilities, and used to compute both a float reference event probability and its stochastic estimate. The principal variable in this stage was the Bernoulli bitstream length N , which was swept across increasing values to evaluate convergence.

The objective of this stage was to answer the most basic architectural question: does the proposed frontend preserve event-level inference quality when heterogeneous channel contributions are encoded stochastically? Detailed implementation and convergence results are provided in the Supplementary Material, Supplement S1 notebook.

5.3. Weighting-Strategy Comparison

The second stage examined weight assignment as a design variable rather than as a fixed constant. Using the same baseline benchmark, four weighting strategies were compared: equal weighting, expert weighting, reliability-aware weighting, data-driven weighting.

For the data-driven case, synthetic event labels were generated from a hidden reference rule, allowing weights to be estimated through logistic-regression-based calibration. The resulting stochastic event estimates were compared against both their own float baselines and a common reference event model.

This stage was designed to determine whether weight assignment materially changes multisensor fusion quality and whether a reliability-aware strategy offers practical advantages over naive equal weighting or the tested data-driven calibration variant. Detailed results are provided in the Supplementary Material, Supplement S2 notebook.

5.4. Static Correlation Study

The third stage investigated the effect of correlation between stochastic bitstreams. Three stream-generation policies were compared: independent, shared, grouped.

In the independent case, each sensor channel was assigned its own randomness source. In the shared case, all channels used a common randomness source. In the grouped case, subsets of channels shared randomness while others remained independent. These policies were tested first in a static setting using full-stream probability estimation.

This stage was introduced to isolate the influence of stochastic correlation structure on multisensor event inference. Because stochastic arithmetic is known to be sensitive to correlation, the objective here was to establish decorrelation as a system-level design requirement. Detailed results are provided in the Supplementary Material, Supplement S3 notebook.

5.5. Time-Varying Synchronization Study

The static correlation study was followed by a time-varying extension in which each channel was represented by a slot-wise probability trajectory $\pi_i(t)$. This second synchronization stage was necessary because simple circular shifts in static full-stream averaging do not strongly affect empirical bitstream means. To expose synchronization effects explicitly, the event estimate was reformulated as a time-resolved quantity $\hat{P}(E,t)$.

The following synchronization modes were then evaluated: perfect synchronization, fixed temporal lags, random temporal jitter.

These were combined with the same three correlation policies: independent, shared, and grouped. This stage was intended to determine whether synchronization mismatch degrades temporal event tracking when multisensor fusion is performed slot-wise rather than through full-stream averaging. The corresponding reproducible implementation is provided in the Supplementary Material, Supplement S3b notebook.

5.6. Robustness Study

The robustness stage evaluated how the frontend behaves under common forms of degradation or misconfiguration. Four robustness categories were examined.

First, additive noise was injected into selected channels at progressively increasing levels.

Second, single-channel dropout was simulated by replacing one channel at a time with a neutral value, allowing channel criticality to be estimated.

Third, random missing-slot corruption was introduced in selected channels, with varying fractions of missing time slots.

Fourth, the assigned fusion weights were perturbed either globally or in a targeted manner, with separate attention to dominant and weak channels.

The purpose of this stage was to determine whether the frontend degrades gracefully and whether different channels contribute equally to the stability of event inference. The complete robustness analyses are given in the Supplementary Material, Supplement S4 notebook.

5.7. Scaling Study

The final experimental stage examined the effect of increasing the number of heterogeneous channels. Systems of 3, 5, 8, 12, and 16 channels were constructed. The added channels were intentionally heterogeneous not only in modality, but also in informational value. Thus, the larger channel sets included not only informative channels, but also weak, redundant, conflicting, and noisy channels.

This design was important because it reflects realistic scaling conditions more faithfully than a purely optimistic expansion with only useful channels. The objective was to determine whether performance continues to improve as the frontend grows and under what conditions mixed-quality channel scaling remains beneficial. These results are also reported in the Supplementary Material, Supplement S4 notebook.

5.8. Evaluation Protocol

Across all stages, the reference event model was computed in the float domain from the mapped channel probabilities and the selected weight configuration. Stochastic event estimates were then generated from Bernoulli bitstreams under the relevant generation, synchronization, robustness, or scaling conditions. The resulting reference and estimated event outputs were compared using: mean absolute error, root mean square error, bias, threshold agreement, peak alignment error in time-varying scenarios.

The use of several complementary metrics was intentional. Average amplitude error alone may understate degradation when threshold decisions or temporal peak localization are affected.

Therefore, the experimental design emphasized joint interpretation of amplitude, decision, and temporal-alignment quality.

5.9. Role of the Supplementary Material in the Experimental Design

Because the present article focuses on architectural interpretation rather than code-level exposition, the experimental workflow was distributed between the main text and a structured supplementary package. The main paper reports the principal scenarios and findings, whereas the Supplementary Material provides the detailed notebook implementations, intermediate plots, parameter sweeps, and reproducibility files. Specifically: Supplement S1 supports baseline validation, Supplement S2 supports weighting analysis, Supplement S3 supports static decorrelation analysis, Supplement S3b supports time-varying synchronization analysis, Supplement S4 supports robustness and scaling analysis.

This structure ensures that the article remains concise while still providing full computational traceability of the reported results.

6. Results

6.1. Baseline Validation of the Universal Frontend

The first result of the study is that the proposed frontend successfully converts heterogeneous sensor channels into a common event-oriented stochastic representation without losing the ability to reproduce the float-reference event estimate. In the baseline five-channel benchmark, the raw sensor channels exhibited clearly different statistical ranges and distributions, confirming that the input layer was genuinely heterogeneous rather than artificially homogenized. After channel-specific mapping, however, all channels were represented in the same normalized event-oriented space $p_i \in [0, 1]$, making them directly compatible with the stochastic fusion stage. Detailed distributions and mapping plots are provided in the Supplementary Material, Supplement S1 notebook.

The float-versus-stochastic comparison also showed that even moderate stream lengths already provide useful inference fidelity. At $N = 256$, the stochastic event estimate was closely aligned with the float baseline, indicating that the interface does not require unrealistically long streams to achieve meaningful event-level agreement. The agreement between the float event model and the stochastic estimate is illustrated in Figure 3.

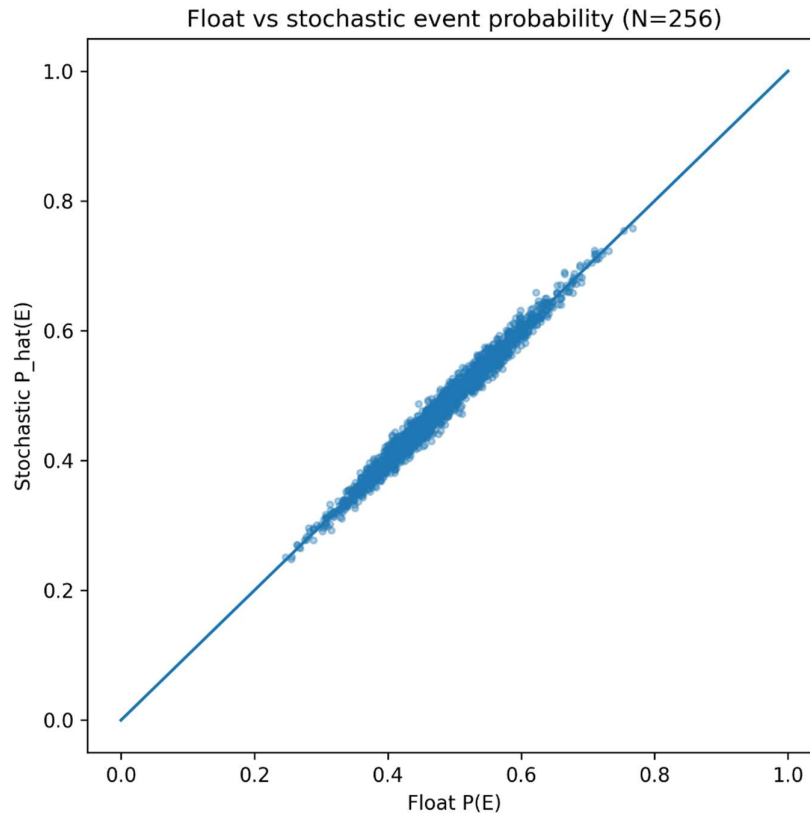
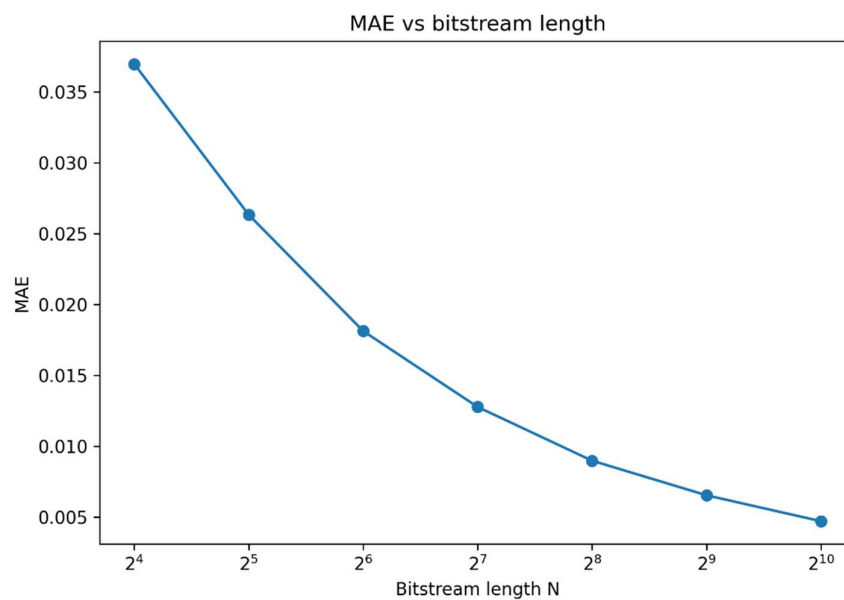


Figure 3. Comparison between the float reference event probability and the stochastic event estimate in the baseline heterogeneous benchmark. The figure illustrates the agreement between deterministic multisensor fusion and its stochastic approximation after channel mapping and Bernoulli bitstream generation.

This is important for practical edge implementations, where latency and bitstream length are constrained resources.

The stochastic event estimate $\hat{P}(E)$ converged toward the float reference $P(E)$ as the Bernoulli bitstream length increased. The convergence of stochastic inference with increasing bitstream length is summarized in Figure 4.



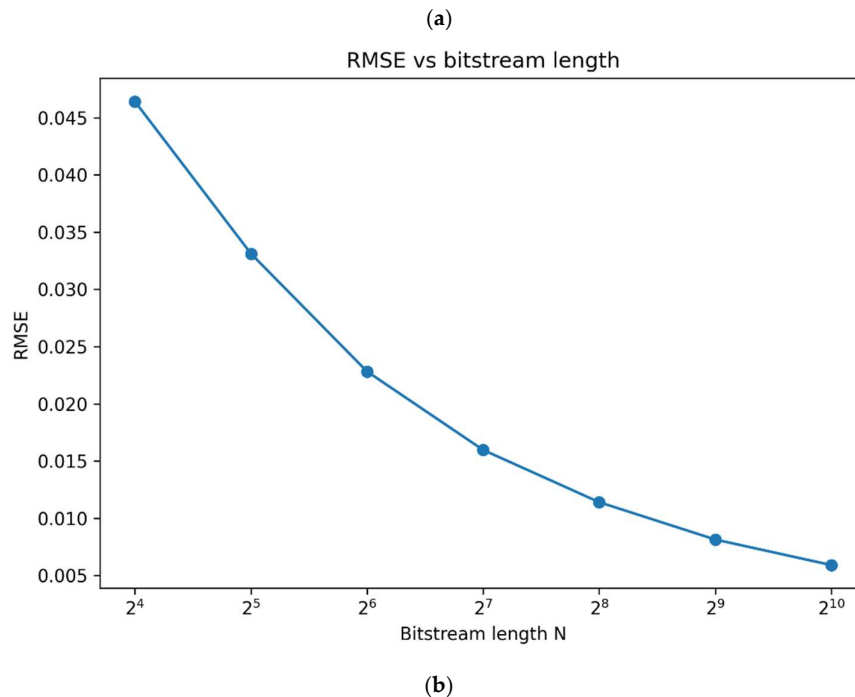


Figure 4. Convergence of stochastic event inference with increasing bitstream length. (a) Mean absolute error (MAE) versus bitstream length. (b) Root mean square error (RMSE) versus bitstream length.

In the baseline validation, the mean absolute error decreased from 0.036966 at $N = 16$ to 0.008969 at $N = 256$, and further to 0.004697 at $N = 1024$. The corresponding RMSE decreased from 0.046415 at $N = 16$ to 0.011401 at $N = 256$, and to 0.005907 at $N = 1024$. These results confirm that the frontend supports controlled approximation of float-domain multisensor inference through stochastic encoding.

Taken together, these results show that the proposed frontend satisfies its most basic requirement: heterogeneous channels can be transformed into a common stochastic interface, and the resulting event estimate converges reliably to the deterministic reference with increasing stochastic precision. This baseline result establishes the frontend as a valid architectural bridge between heterogeneous sensing and stochastic fusion. Additional convergence plots and implementation details are given in the Supplementary Material, Supplement S1 notebook.

6.2. Weight Assignment as a Design Variable

The second major result is that weight assignment is a first-order design variable of the frontend rather than a secondary tuning detail. Using the same five-channel benchmark, four weighting strategies were compared: equal, expert, reliability-aware, and data-driven. A comparison of the tested weighting strategies is presented in Figure 5.

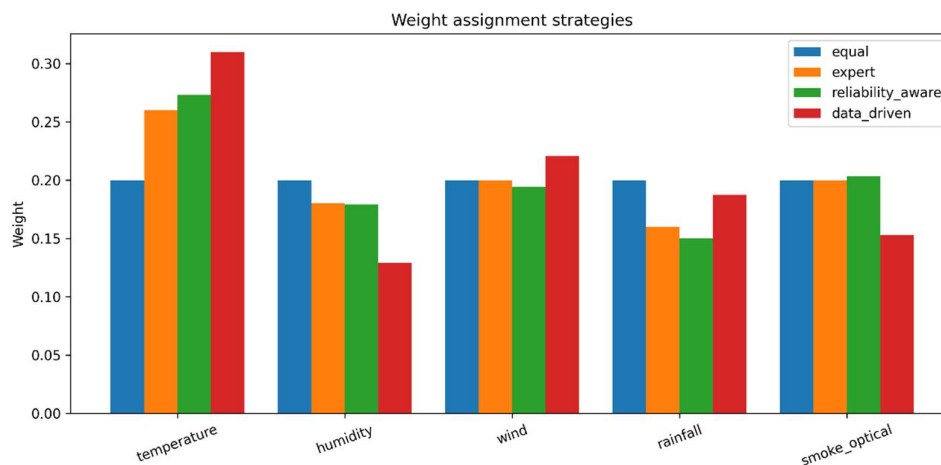


Figure 5. Comparison of weight-assignment strategies for event-level stochastic fusion. The figure summarizes the behavior of equal, expert, reliability-aware, and tested data-driven weighting, highlighting the effect of weight selection on inference accuracy and threshold consistency.

The comparison showed that naive equal weighting was consistently inferior to the better-structured alternatives, while reliability-aware weighting provided the best overall performance in the tested benchmark.

At $N=256$, the stochastic estimate produced with equal weights yielded a MAE of 0.071738 relative to the reference event model, whereas expert weights reduced the MAE to 0.056903. Reliability-aware weighting improved it further to 0.053950, while the tested data-driven weighting variant reached 0.070957. The same ordering was observed in RMSE and threshold agreement. In particular, threshold agreement at $N=256$ was 0.760667 for equal weighting, 0.833000 for expert weighting, 0.844667 for reliability-aware weighting, and 0.761000 for the tested data-driven variant. These results indicate that incorporating channel reliability into the weighting rule improves not only average error but also decision consistency.

This advantage remained visible at shorter stream lengths, which is especially relevant for resource-constrained stochastic inference. At $N=16$, the MAE was 0.076646 for equal weighting, 0.064345 for expert weighting, and 0.062157 for reliability-aware weighting, again preserving the same ranking. Thus, the benefit of structured weighting was not limited to high-precision stochastic settings.

The robustness comparison under a noisy smoke-optical channel provided an additional perspective. Under this perturbation, reliability-aware weighting again produced the best results, with lower MAE and RMSE than equal, expert, or the tested data-driven weighting. This suggests that reliability-aware weighting is not only accurate under nominal conditions but also more stable under channel degradation.

The tested data-driven weighting strategy did not outperform the reliability-aware alternative in the present synthetic benchmark. This should be interpreted cautiously: the result reflects the specific benchmark setup and the particular calibration pipeline used here, not a general rejection of data-driven weighting. Nevertheless, it shows that a structured reliability-aware scheme can already provide strong practical performance without requiring a large-scale training pipeline. Full comparative tables and plots are reported in the Supplementary Material, Supplement S2 notebook.

Overall, the weighting study demonstrates that the frontend is not just a mapping layer. It is a controlled fusion architecture in which the semantics, trustworthiness, and relative importance of heterogeneous channels must be encoded explicitly. In the tested benchmark, reliability-aware weighting provided the best balance of accuracy, interpretability, and robustness.

6.3. Correlation and Synchronization Effects

The third major result concerns the role of stochastic correlation and synchronization. In the static stochastic fusion setting, the quality of event inference depended strongly on the inter-channel correlation structure of the Bernoulli streams, in agreement with the known sensitivity of stochastic computing to correlated bitstreams [1]. The effect of independent, grouped, and shared-randomness stream generation on inference quality is shown in Figure 6.

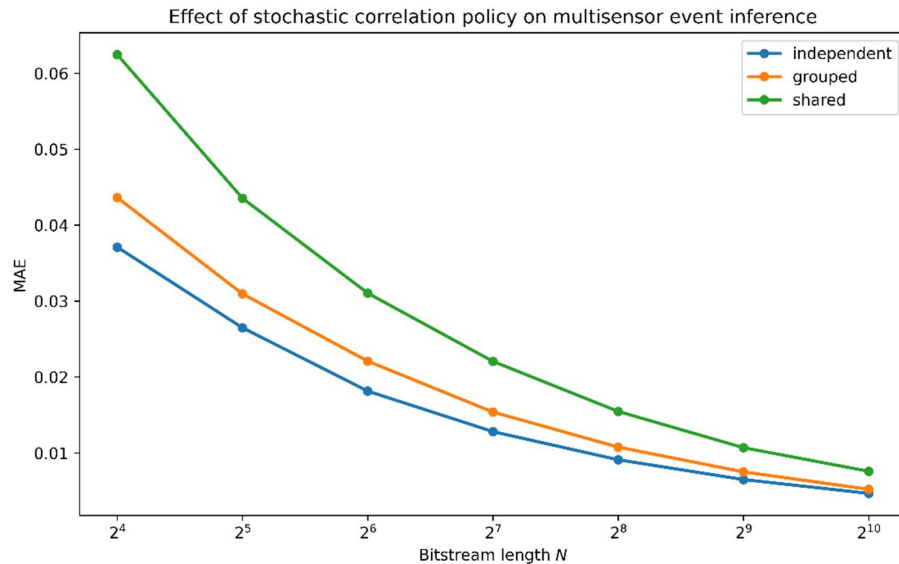


Figure 6. Effect of stochastic correlation policy on multisensor event inference. Mean absolute error (MAE) is shown as a function of bitstream length for independent, grouped, and shared-randomness stream generation under the perfectly synchronized condition.

Independent stream generation produced the best results, grouped generation produced intermediate results, and shared-randomness generation produced the worst results.

At $N=256$, the independent policy yielded MAE = 0.009094, RMSE = 0.011500, and threshold agreement = 0.960000. Under grouped correlation, the corresponding values were MAE = 0.010764, RMSE = 0.013485, and threshold agreement = 0.950333. Under shared randomness, performance degraded to MAE = 0.015462, RMSE = 0.019574, and threshold agreement = 0.927000. This ranking was preserved across the tested stream lengths. The effect was particularly pronounced at small N : at $N=16$, shared randomness produced markedly higher error and lower threshold agreement than the independent policy. These results establish decorrelation as a system-level requirement for accurate stochastic multisensor fusion. Full static correlation results are given in the Supplementary Material, Supplement S3 notebook.

The static synchronization experiments, however, showed only negligible sensitivity to circular bitstream shifts. This was an expected limitation of the static formulation, because full-stream probability estimation depends mainly on empirical means, which are invariant under circular permutation. This observation motivated the introduction of the time-varying formulation.

In the time-varying slot-wise setting, synchronization became a measurable factor. Time-varying synchronization effects under lag and jitter conditions are reported in Figure 7.

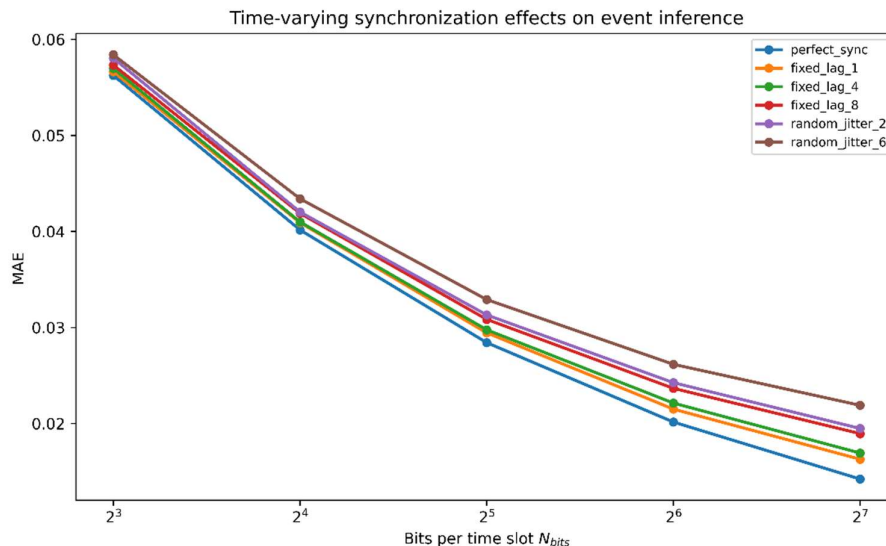


Figure 7. Time-varying synchronization effects on multisensor event inference. Mean absolute error (MAE) is shown as a function of bits per time slot for the independent stochastic generation policy under perfect synchronization, fixed lag, and random jitter conditions.

Once each channel was represented by a temporal probability trajectory $p_i(t)$ and event inference was performed slot by slot, temporal misalignment between channels degraded event tracking quality. Under independent stream generation, the best result was obtained in the perfectly synchronized case, with MAE = 0.020144, RMSE = 0.025274, threshold agreement = 0.985234, and peak alignment error = 11.9425. As synchronization mismatch increased, performance degraded gradually. For example, under independent generation, MAE increased from 0.020144 in the perfect synchronization case to 0.023654 under fixed_lag_8.

The same qualitative behavior was observed for grouped generation, while shared-randomness policies remained inferior overall. In the time-varying benchmark, grouped perfect synchronization gave MAE = 0.025063, RMSE = 0.031457, and threshold agreement = 0.978398, whereas shared perfect synchronization yielded MAE = 0.037996, RMSE = 0.047630, and threshold agreement = 0.948730. Thus, both correlation structure and synchronization quality affected temporal event inference, with independent synchronized streams providing the best overall behavior.

The peak alignment error proved especially informative in the time-varying setting. It captured degradation in temporal event localization that was not always fully reflected by amplitude-based metrics alone. The metric decreased as the number of bits per time slot increased, but remained systematically worse for grouped and shared policies than for independent generation. This confirms that synchronization and decorrelation are not merely implementation details; they directly affect the frontend's ability to track event dynamics in time.

Overall, the correlation and synchronization results support two distinct but complementary conclusions. First, in static stochastic fusion, inter-stream decorrelation is a first-order requirement. Second, in time-varying fusion, synchronization becomes an additional first-order factor, because temporal event inference is sensitive to slot misalignment. Detailed plots, heatmaps, and temporal tracking examples are provided in the Supplementary Material, Supplements S3 and S3b.

6.4. Robustness under Channel Degradation and Weight Mismatch

The robustness experiments showed that the proposed frontend degrades gracefully under several classes of perturbations, although different channels and design variables affect the system unequally. The robustness of the frontend under channel degradation and weight perturbation is summarized in Figure 8.

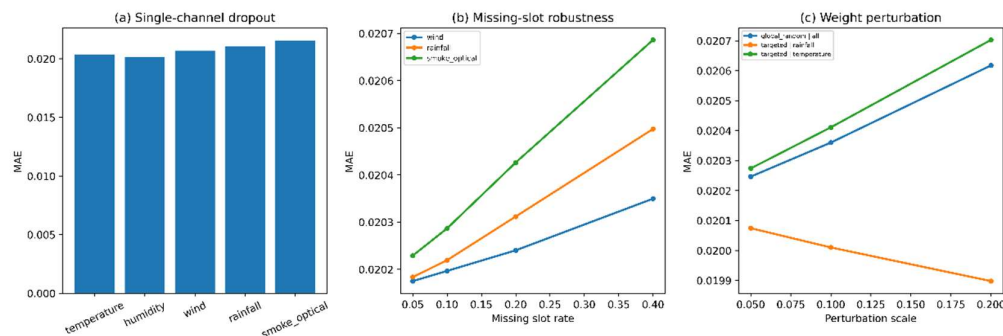


Figure 8. Robustness of the proposed frontend under representative degradation and misconfiguration scenarios. (a) Sensitivity to single-channel dropout. (b) Event-inference degradation under missing-slot corruption. (c) Sensitivity to weight perturbation under global and targeted mismatch conditions.

This non-uniform behavior is itself an important architectural result, because it reveals which channels are critical and which perturbations are most damaging.

In the additive-noise study, the average amplitude error did not always increase monotonically with noise severity. In some cases, MAE decreased slightly as noise increased. For example, in the temperature-noise scenario, the baseline MAE of 0.020144 changed to 0.018756 at the highest tested noise level. Similar mild decreases were observed in the humidity and smoke-optical noise scenarios. However, this should not be interpreted as genuine improvement of inference quality. The more decision-oriented metrics showed the opposite tendency: threshold agreement decreased and peak alignment error increased under stronger noise. For instance, with strong noise in the temperature channel, threshold agreement dropped from 0.985234 to 0.953164, while peak alignment error increased beyond 13 slots. This indicates that amplitude-only metrics can understate degradation when noise blurs decision boundaries or temporal peak localization.

The channel-dropout study provided a clearer picture of channel criticality. Among the five baseline channels, smoke_optical was the most critical: replacing it with a neutral value increased MAE to 0.021550, reduced threshold agreement to 0.926582, and increased peak alignment error to 17.99. Rainfall was the next most critical, whereas humidity had the smallest effect on overall performance. These results show that the frontend does not treat all channels equally in practice, even if they are all formally represented in the same stochastic format. Channel-dropout analysis therefore provides a meaningful way to identify dominant evidence sources in a heterogeneous event-inference system.

The missing-slot study further demonstrated graceful degradation. As the fraction of missing slots increased, MAE rose gradually rather than catastrophically. Again, smoke_optical was among the most sensitive channels, while wind was less sensitive. Thus, the system preserves functionality under partial data loss, but performance depends on which channel is degraded and how informative that channel is for the event.

Weight perturbation experiments revealed that errors in dominant channels are more harmful than errors in weaker ones. Targeted perturbation of the strongest informative channel produced greater degradation than comparable perturbation of a weaker channel. This result is consistent with the weighting study and reinforces the interpretation that the frontend is most sensitive to miscalibration where informative channels carry a large share of the fusion burden.

Together, these robustness results support four points. First, the frontend is not fragile: it degrades progressively under realistic corruption scenarios. Second, channel criticality is non-uniform and can be quantified. Third, robustness must be assessed with several metrics simultaneously, because MAE alone may obscure threshold or temporal degradation. Fourth, dominant informative channels require the most careful calibration and protection. Full robustness tables and plots are reported in the Supplementary Material, Supplement S4 notebook.

6.5. Scaling Behavior Under Increasing Channel Count

The scaling study showed that, in the tested benchmark, increasing the number of channels from 3 to 16 improved event-inference quality despite the inclusion of weak, redundant, conflicting, and noisy channels. The effect of increasing channel count on event-inference quality is shown in Figure 9.

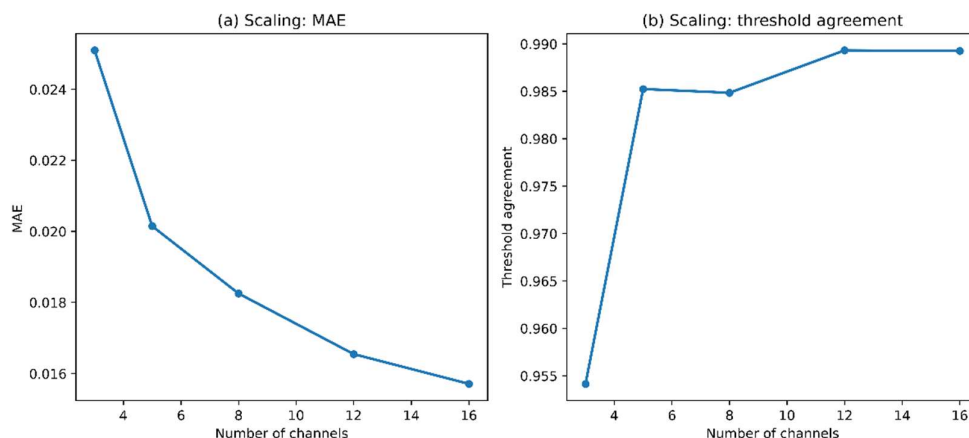


Figure 9. Scaling behavior of the universal frontend with increasing heterogeneous channel count. (a) Mean absolute error (MAE) as a function of the number of channels. (b) Threshold agreement as a function of the number of channels. The results show favorable scaling in the tested benchmark despite the inclusion of mixed informative, weak, redundant, conflicting, and noisy channels.

This result is important because it suggests that the proposed frontend can benefit from heterogeneous expansion, provided that the added channel set still contains sufficient useful structure.

With 3 channels, the frontend achieved MAE = 0.025091 and threshold agreement = 0.954141. Expanding to the 5-channel baseline reduced MAE to 0.020144 and increased threshold agreement to 0.985234. Further expansion to 8, 12, and 16 channels reduced MAE to 0.018249, 0.016544, and 0.015707, respectively, while threshold agreement remained high and even increased slightly, reaching 0.989258 in the 16-channel case. Peak alignment error also improved from 11.9425 in the 5-channel baseline to 10.5400 and 10.4525 in the 12- and 16-channel configurations.

This trend should be interpreted carefully. The result does not imply that adding channels always improves event inference in all systems. Rather, in the present benchmark, the added channels did not overwhelm the fusion process with useless or contradictory information; enough informative structure remained dominant that the additional evidence improved the aggregate event estimate. In other words, the scaling advantage depended on the mixture of channel types and on the frontend's ability to absorb heterogeneous information through weighting and stochastic fusion.

This is an important practical conclusion. A universal frontend should not be evaluated only in minimal sensor configurations; it must also be assessed under realistic growth in channel count and diversity. The present results indicate that the proposed architecture scales favorably under the tested benchmark conditions, while also preserving robustness to mixed channel quality. Detailed scaling configurations and full metric tables are provided in the Supplementary Material, Supplement S4 notebook.

6.6. Summary of Main Findings

Across all experimental stages, the results support a coherent architectural interpretation of the frontend. First, heterogeneous sensors can be mapped into a common event-oriented stochastic representation with controllable convergence to a float reference. Second, weight assignment materially affects fusion quality, with reliability-aware weighting emerging as the strongest tested

strategy in the current benchmark. Third, decorrelation is essential for accurate stochastic multisensor fusion, and synchronization becomes essential once the event model is explicitly time-varying. Fourth, the frontend exhibits graceful degradation under noise, dropout, missing data, and weight perturbation, while remaining sensitive to channel criticality. Finally, scaling can improve inference quality when the added heterogeneous channels still preserve sufficient informative structure.

These findings collectively support the main thesis of the paper: the proposed universal edge frontend is not merely a signal-format converter, but a structured multisensor architecture whose performance is governed by event-oriented mapping, stochastic precision, weighting, decorrelation, synchronization, robustness, and scaling. Detailed computational evidence for each of these claims is provided in the Supplementary Material, Supplements S1–S4.

7. Discussion

The results support the interpretation of the proposed framework as a universal event-oriented frontend rather than as a task-specific fusion script or a signal-representation module. The key architectural idea is that heterogeneous sensor channels do not need to be made identical in their raw physical units before entering a photonic stochastic system. Instead, they can be translated at the edge into normalized contributions to an event of interest, and only then fused in a common stochastic domain. This shift is important because it aligns the frontend with the actual inference objective: in many practical sensing systems, the desired output is not waveform reconstruction but estimation of a risk, event, anomaly, or decision state.

A first important implication of the study is that event-level unification is sufficient for meaningful multisensor interfacing. The baseline results showed that strongly heterogeneous channels can be mapped into a shared probability space and then encoded into Bernoulli bitstreams while preserving convergence toward a float reference model. This means that the frontend can act as a representation bridge between diverse sensor modalities and a downstream stochastic processor without requiring a conventional feature-space homogenization pipeline. In architectural terms, the frontend transforms heterogeneity from a raw-data incompatibility problem into a controlled event-probability mapping problem.

A second important implication is that weighting, decorrelation, and synchronization must be treated as primary design variables, not secondary implementation details. The weighting study showed that naive equal weighting is inadequate when channels differ in reliability and inferred relevance, and that a reliability-aware strategy can outperform both equal weighting and the tested data-driven calibration variant in the current benchmark. This suggests that even before introducing complex learning pipelines, practical multisensor systems can gain measurable benefit from explicitly modeling channel trustworthiness in the fusion rule. For a journal such as *Sensors*, this is an important systems-level message: the quality of event inference depends not only on sensor hardware and not only on the downstream classifier, but also on how the interface interprets and weights each channel.

The decorrelation results reinforce a classical but often underemphasized property of stochastic computing: shared randomness is harmful when multiple channels are intended to contribute independent evidence. In the present study, independent stream generation consistently produced the best event-inference accuracy, grouped generation gave intermediate performance, and shared-randomness generation performed worst. This result is especially relevant because a universal frontend could otherwise appear to be a purely semantic mapping layer. The experiments show that it is more than that: the frontend is also responsible for preserving statistical independence when the fusion logic requires it. In other words, the interface is not only about what probabilities are assigned to channels, but also about how those probabilities are physically or algorithmically realized as stochastic streams.

The time-varying synchronization study adds another layer of practical relevance. In the static setting, synchronization offsets had little visible effect because full-stream averages are insensitive to circular reordering. However, once inference was reformulated as a time-resolved slot-wise process,

synchronization became a measurable factor. This distinction is conceptually useful. It shows that synchronization should not be discussed in the abstract, but in relation to the temporal semantics of the target task. If the goal is only average event probability over a long window, synchronization may be less critical. If the goal is event tracking in time, then lag and jitter directly degrade performance. The present results therefore suggest that synchronization requirements depend on whether the stochastic frontend supports static decision support or dynamic event monitoring.

The robustness analysis provides an additional systems-level message: the frontend exhibits graceful degradation, but not all degradations are equivalent. Channel dropout, missing-slot corruption, and weight mismatch do not affect all channels equally. In the tested benchmark, the optical smoke-related channel emerged as one of the most critical evidence sources, while other channels were less decisive. This non-uniformity is not a weakness; rather, it is valuable diagnostic information. It means that robustness studies can be used not only to assess failure tolerance, but also to identify the channels that deserve stronger protection, calibration, redundancy, or reliability-aware weighting in a real deployment. Similarly, the weight-perturbation study showed that errors in dominant informative channels are more harmful than comparable errors in weak channels, which is highly relevant for practical calibration workflows.

One of the more subtle findings of the robustness stage is that different metrics capture different failure modes. In particular, additive noise did not always increase MAE, yet it often reduced threshold agreement and worsened peak alignment. This indicates that average amplitude fidelity is not sufficient for evaluating event-level frontends. A system may appear stable under one metric while already losing reliability in thresholded decisions or temporal localization. For that reason, the joint use of MAE, RMSE, threshold agreement, and peak alignment error is not only methodologically justified but necessary. This point may also help position the paper within the sensor-systems literature, where average regression metrics alone are sometimes insufficient to capture operational decision quality.

The scaling results are encouraging, but they should be interpreted carefully. In the tested benchmark, increasing the number of channels improved performance even when the added set included weak, redundant, conflicting, and noisy channels. This suggests that the proposed frontend can exploit heterogeneous expansion effectively when useful structure remains dominant. However, this should not be overgeneralized into the claim that adding channels is always beneficial. In real systems, the value of scaling will depend on the ratio of informative to misleading inputs, on the quality of weighting, and on whether the frontend can suppress harmful correlations and misalignment. Thus, the present result is best viewed as evidence that the architecture can scale favorably under mixed-quality conditions, not as proof of unconditional monotonic benefit.

These findings also help clarify the intended scope of the paper relative to neighboring research directions. The work is not centered on compact stochastic encoding of a single waveform, harmonic factorization, or reconstruction fidelity. Those are valid topics, but they address a different problem. The contribution here is architectural: a universal edge stochastic frontend for heterogeneous sensors, oriented toward event inference and photonic compatibility. The frontend is evaluated not only by representational convergence, but by the system-level factors that determine whether such an interface is practically useful: weighting, decorrelation, synchronization, robustness, and scaling.

At the same time, several limitations should be stated clearly. First, the present study is computational and benchmark-driven; it does not report a full hardware photonic prototype. The polarization-compatible optical interface is introduced as a physically meaningful interface abstraction, not as a complete experimental implementation. Second, the sensor scenarios are synthetic and architecture-oriented rather than derived from a fully instrumented real dataset. This was intentional, because the goal was to isolate frontend design variables in a controlled manner. Third, the tested data-driven weighting strategy is only one calibration variant, and stronger learning-based weighting methods may yield different rankings in future work. Finally, the time-varying benchmark, while sufficient to expose synchronization effects, remains simplified compared with real-world asynchronous multisensor dynamics.

These limitations point naturally to future work. A next step would be to instantiate the proposed interface in a concrete polarization-based photonic pipeline, including modulators, detectors, and synchronization circuitry. Another direction would be to validate the frontend on real heterogeneous sensing datasets, especially in event-detection domains where channel reliability varies over time. Context-adaptive weighting, online recalibration, and joint optimization of edge mapping with downstream stochastic fusion are also promising extensions. More broadly, the present results suggest that the proposed frontend can serve as a design template for photonic edge-inference systems in which heterogeneous sensing must be integrated through a common probabilistic interface rather than through raw signal transport.

Overall, the discussion supports a clear conclusion: the proposed architecture is meaningful precisely because it treats the sensor interface as an inference-aware stochastic system. By shifting the edge representation from raw heterogeneous measurements to unified event-oriented stochastic contributions, it creates a practical pathway between multisensor acquisition and future photonic stochastic processors.

8. Conclusions

This paper proposed a universal edge frontend for heterogeneous sensors in photonic stochastic systems. The central idea was to convert physically different sensor measurements into a common event-oriented probability space and then represent these contributions as Bernoulli bitstreams compatible with a polarization-based optical interface. In this way, the frontend serves as a bridge between heterogeneous sensing and stochastic photonic processing without requiring direct fusion in raw physical units.

The results showed that the proposed frontend provides controlled convergence from stochastic event inference to a float reference model as the bitstream length increases. This confirms that heterogeneous channels can be unified into a common stochastic event representation while preserving useful inference fidelity.

The study also demonstrated that weighting strategy is a key design variable of the frontend. In the tested benchmark, reliability-aware weighting provided the best overall balance of accuracy, threshold consistency, and robustness, outperforming naive equal weighting and the tested data-driven weighting variant.

A further important conclusion is that stochastic stream decorrelation and temporal synchronization must be treated as core architectural constraints. Independent bitstream generation consistently produced better event-inference quality than grouped or shared-randomness policies, while synchronization mismatch became clearly measurable in the time-varying formulation, where it degraded temporal event tracking.

The robustness and scaling studies showed that the frontend degrades gracefully under channel corruption, missing data, and weight perturbation, while also exhibiting non-uniform channel criticality. In the tested heterogeneous benchmark, scaling from smaller to larger channel sets improved inference quality, indicating that the proposed architecture can benefit from structured multisensor expansion when informative contributions remain dominant.

Overall, the presented framework establishes a practical event-level sensor-to-stochastic interface for photonic systems. The results suggest that heterogeneous sensing can be coupled to future photonic stochastic processors through a unified frontend in which weighting, decorrelation, synchronization, robustness, and scaling are treated as first-order system design variables.

Supplementary Materials: The following supporting information can be downloaded at the website of this paper posted on Preprints.org.

Author Contributions: Conceptualization, M.S.; methodology, M.S.; software, M.S.; validation, M.S., R.Z. and J.C.; formal analysis, M.S.; investigation, M.S.; resources, O.V.; data curation, M.S.; writing—original draft preparation, M.S.; writing—review and editing, O.V., C.Z., X.Z. and J.C.; visualization, M.S.; supervision, O.V.;

project administration, O.V.; funding acquisition, J.Z. All authors have read and agreed to the published version of the manuscript.”.

Funding: This research received no external funding. Open-access publication charges were covered by the Taizhou Institute of Zhejiang University, Taizhou, China.

Institutional Review Board Statement: This study did not involve human participants, human data, or animal subjects, and therefore did not require ethical approval.

Data Availability Statement: All data supporting the reported results are included in the article and in the Supplementary Material. The Supplementary Material contains the reproducible computational notebooks, scripts, generated result files, and supporting figures and tables used in the present study. The results are based on computationally generated benchmark scenarios prepared specifically for this work; no external experimental dataset was used as the primary source of the reported findings.

Acknowledgments: The authors thank colleagues associated with the NRFU project No. 2025.07/0069 (“Recent applications of structured optical fields in polarization-interference methods for solving problems in telecommunications and biomedicine”) for helpful discussions and non-financial support.

Conflicts of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MAE	Mean Absolute Error
$RMSE$	Root Mean Square Error
SC	Stochastic Computing
$P - bit$	Probabilistic bit
H/V	Horizontal/Vertical polarization states
CSV	Comma-Separated Values
$JSON$	JavaScript Object Notation
DOI	Digital Object Identifier
$P(E)$	Float reference event probability
$\hat{P}(E)$	Stochastic estimate of event probability
$P(E, t)$	Time-varying reference event probability
$\hat{P}(E, t)$	Time-varying stochastic estimate of event probability
p_i	Event-oriented probability contribution of sensor channel i
\hat{p}_i	Bitstream-based estimate of channel probability
$p_i(t)$	Time-varying event-oriented probability contribution of channel i
$\hat{p}_i(t)$	Slot-wise estimate of channel probability
S_i	Bernoulli bitstream generated from channel probability p_i
$S_i(t)$	Time-varying Bernoulli bitstream
w_i	Fusion weight of sensor channel i
θ	Decision threshold
α	Sigmoid slope parameter
N	Bitstream length
N_{bits}	Bits per time slot
T	Number of time slots

References

1. Li, X.; Dunkin, F.; Dezert, J. Multi-source information fusion: Progress and future. *Chin. J. Aeronaut.* 2024, 37(7), 24–58. <https://doi.org/10.1016/j.cja.2023.12.009>
2. Shaikh, Z.A.; Van Hamme, D.; Veelaert, P.; Philips, W. Probabilistic fusion for pedestrian detection from thermal and colour images. *Sensors* 2022, 22(22), 8637. <https://doi.org/10.3390/s22228637>
3. Rabb, E.; Steckenrider, J.J. Walking trajectory estimation using multi-sensor fusion and a probabilistic step model. *Sensors* 2023, 23(14), 6494. <https://doi.org/10.3390/s23146494>
4. Qian, H.; Wang, M.; Zhu, M.; Wang, H. A review of multi-sensor fusion in autonomous driving. *Sensors* 2025, 25(19), 6033. <https://doi.org/10.3390/s25196033>
5. Alaghi, A.; Hayes, J.P. Survey of stochastic computing. *ACM Trans. Embed. Comput. Syst.* 2013, 12(2s), 92:1–92:19. <https://doi.org/10.1145/2465787.2465794>
6. Camsari, K.Y.; Sutton, B.M.; Datta, S. p-bits for probabilistic spin logic. *Appl. Phys. Rev.* 2019, 6(1), 011305. <https://doi.org/10.1063/1.5055860>
7. Xu, X.; Tan, M.; Corcoran, B.; Wu, J.; Boes, A.; Nguyen, T.G.; Chu, S.T.; Little, B.E.; Hicks, D.G.; Morandotti, R.; Mitchell, A.; Moss, D.J. 11 TOPS photonic convolutional accelerator for optical neural networks. *Nature* 2021, 589, 44–51.
8. Dong, B.; Aggarwal, S.; Zhou, W.; Ali, U.E.; Farmakidis, N.; Lee, J.S.; He, Y.; Li, X.; Kwong, D.-L.; Wright, C.D.; Pernice, W.H.P.; Bhaskaran, H. Higher-dimensional processing using a photonic tensor core with continuous-time data. *Nat. Photonics* 2023, 17, 1080–1088.
9. Wu, J.; Lin, X.; Guo, Y.; Liu, J.; Fang, L.; Jiao, S.; Dai, Q. Analog optical computing for artificial intelligence. *Engineering* 2022, 10, 133–145. <https://doi.org/10.1016/j.eng.2021.06.021>
10. Xu, D.; Ma, Y.; Jin, G.; Cao, L. Intelligent photonics: A disruptive technology to shape the present and redefine the future. *Engineering* 2025, 46, 186–213.
11. Baek, Y.; Bae, B.; Shin, H.; Sonnadara, C.; Cho, H.; Lin, C.-Y.; Mu, Y.; Shen, C.; Shah, S.; Wang, G.; Lee, K. Edge intelligence through in-sensor and near-sensor computing for the artificial intelligence of things. *npj Unconventional Computing* 2025, 2, 25.
12. Zhang, S.; Jiang, X.; Wu, B.; Zhou, H.; Xu, W.; Zhou, H.; Ruan, Z.; Dong, J.; Zhang, X. Photonic edge intelligence chip for multi-modal sensing, inference and learning. *Nat. Commun.* 2025, 16, 10136.
13. Xue, H.; Zhang, M.; Yu, P.; Zhang, H.; Wu, G.; Li, Y.; Zheng, X. A novel multi-sensor fusion algorithm based on uncertainty analysis. *Sensors* 2021, 21(8), 2713. <https://doi.org/10.3390/s21082713>
14. Roheda, S.; Krim, H.; Luo, Z.-Q.; Wu, T. Event driven sensor fusion. *Signal Process.* 2021, 188, 108241. <https://doi.org/10.1016/j.sigpro.2021.108241>
15. Gutiérrez, R.; Rampérez, V.; Paggi, H.; Lara, J.A.; Soriano, J. On the use of information fusion techniques to improve information quality: Taxonomy, opportunities and challenges. *Inf. Fusion* 2022, 78, 102–137. <https://doi.org/10.1016/j.inffus.2021.09.017>
16. Mao, W.-L.; Wang, C.-C.; Chou, P.-H.; Liu, K.-C.; Tsao, Y. MECKD: Deep learning-based fall detection in multilayer mobile edge computing with knowledge distillation. *IEEE Sens. J.* 2024, 24(24), 42195–42209. <https://doi.org/10.1109/JSEN.2024.3456577>
17. Angelsky, O.V., Bekshaev, A.Y., Zenkova, C.Y., Ivansky, D.I., Zheng, J. (2022). Correlation Optics, Coherence and Optical Singularities: Basic Concepts and Practical Applications, *Frontiers in Physics*, 10, 924508. <https://doi.org/10.3389/fphy.2022.924508>
18. O. Angelsky, M. Strynadko, C. Zenkova, R. Zaiats, Zhang Xinzhen, Jun Zheng, and Jingxian Cai. Supplementary materials for “ Universal Sensor Frontend for Event Inference in Photonic Stochastic Systems”. figshare (2026). <https://doi.org/10.6084/m9.figshare.31855165>

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.