NÖHÜ Müh. Bilim. Derg. / NOHU J. Eng. Sci., Erken Görünüm / InPress

Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi
Niğde Ömer Halisdemir University Journal of Engineering Sciences

Derleme makalesi / Review article

www.dergipark.org.tr/tr/pub/ngumuh / www.dergipark.org.tr/en/pub/ngumuh

Taxonomy of sensor fusion techniques for various application areas: A review

Çeşitli uygulama alanları için sensör füzyon tekniklerinin taksonomisi: Bir inceleme

Hande Çavşi Zaim[1,*], Esra N. Yolaçan[2]

[1,2] *Eskişehir Osmangazi Üniversity, Department of Computer Engineering, 26040 Eskişehir, Türkiye*

Abstract

Sensor fusion techniques play critical roles in various industries such as defense, automotive, military, and healthcare. These techniques combine data from multiple sources, resulting in more detailed and reliable results. Sensor fusion techniques are indispensable for effective decision-making processes, especially in complex environments and variable conditions. These techniques allow systems to operate more efficiently. This study examines the advantages, challenges, and different algorithms used in various sensor fusion techniques and provides a comprehensive classification. This classification makes it possible to evaluate sensor fusion techniques and categorize them to appeal to broader applications. The study aims to help researchers understand sensor fusion techniques and guide them in making choices that suit their needs. Additionally, when evaluating the future potential of sensor fusion, the focus is on how fusion techniques may evolve, particularly with increasing complexity and diversity. Thus, it contributes to advancing research in sensor fusion and developing more effective systems.

Keywords: Data fusion, Information fusion, Sensor fusion, Sensor fusion algorithms, Sensor fusion taxonomy

Öz

Sensör füzyon teknikleri savunma, otomotiv, askeri ve sağlık gibi çeşitli endüstrilerde kritik rol oynamaktadır. Bu teknikler, birden fazla kaynaktan gelen verileri birleştirerek daha ayrıntılı ve güvenilir sonuçların elde edilmesini sağlar. Özellikle karmaşık ortamlarda ve değişken koşullarda etkili karar verme süreçleri için vazgeçilmez olan sensör füzyon teknikleri sistemlerin daha verimli çalışmasına olanak tanır. Bu çalışma, çeşitli sensör füzyon tekniklerinin avantajlarını, zorluklarını ve kullanılan farklı algoritmaları detaylı bir şekilde incelemekte ve kapsamlı bir sınıflandırma sunmaktadır. Bu sınıflandırma, çeşitli sensör füzyon tekniklerini değerlendirmeyi ve bunları daha geniş bir uygulama alanına hitap edecek şekilde kategorilere ayırmayı mümkün kılar. Çalışmanın amacı araştırmacılara sensör füzyon tekniklerini daha iyi anlamalarını sağlamak ve ihtiyaçlarına uygun seçimler yapmaları için rehberlik etmektir. Ayrıca, sensör füzyonunun gelecekteki potansiyeli değerlendirilirken, özellikle artan karmaşıklık ve çeşitlilikle birlikte füzyon tekniklerinin nasıl gelişebileceğine değinilmiştir. Böylece sensör füzyonunda araştırmaların ilerlemesine ve daha etkili sistemlerin geliştirilmesine katkı sağlanır.

Anahtar kelimeler: Bilgi füzyonu, Sensör füzyonu, Sensör füzyonu algoritmaları, Sensör füzyonu taksonomisi, Veri füzyonu.

1 Introduction

The Joint Directors of Laboratories (JDL) committee determined the definition of sensor fusion as follows: Sensor fusion or information fusion is a multi-level procedure that deals with the association, correlation, and integration of data and information from single and multiple sources to obtain distinctive locations and determine predictions [1]. When the studies in the literature are examined, it has been observed that the terms sensor fusion, information fusion, and data fusion are used interchangeably. *Information fusion techniques* combine data from multiple sensors and related databases to achieve improved accuracy and more specific definitions compared to a single sensor [2]. *Sensor fusion* is the collaborative use of information provided by multiple sensors to help perform a function [3,4]. *Data fusion* combines data from various sources to improve system performance [5-7]. Regardless of the different definitions in the literature, sensor fusion can be summarised as integrating information from multiple sources to increase the accuracy and quality of the content and reduce cost.

Sensor fusion technology plays a critical role in many industries, such as automation, robotics, and artificial intelligence. This technology integrates data from different sources, enabling more comprehensive, reliable and in-depth analysis. In this way, correct and timely decisions can be made in complex environments and changing conditions, system performance can be increased, and innovative solutions can be developed. Sensor fusion has a wide range of applications, from industrial processes to healthcare, defence technologies, and environmental monitoring systems. Sensor fusion techniques direct future technological developments by providing advanced data integration and analytics in these areas.

The advantages of sensor fusion techniques can be listed as 1) reduction in uncertainty, 2) increase in accuracy, and 3)

* Sorumlu yazar / Corresponding author, e-posta / e-mail: handecavsi43@gmail.com (H. Çavşi Zaim)
Geliş / Recieved: 19.03.2024 Kabul / Accepted: 02.08.2024 Yayımlanma / Published: xx.xx.20xx
doi: 10.28948/ngumuh.1455086

cost reduction [8]. Sensor fusion techniques can be considered the most suitable method for achieving a certain level of accuracy in integrating multiple sensors, as the inadequacy of data from a single sensor can be compensated for by data from other sensors.

Sensor fusion applications can effectively solve many problems in various areas. The main areas where sensor fusion applications are used include the Internet of Things (IoT), automotive and navigation, quadrotors and drones, computer vision, virtual reality/augmented reality, and healthcare [9].

Sensor fusion helps provide context awareness of IoT. Context awareness is the ability of a system or device to detect environmental conditions, user situations, or contexts of interaction. Sensor fusion provides more comprehensive and reliable information by combining data from different sensors, thus enabling systems and devices to detect interaction contexts more accurately and quickly. The number of IoT devices and the data types they collect are increasing daily, making sensor fusion techniques even more critical. For example, sensor fusion techniques are needed in IoT fields such as smart energy consumption control [10], power grid [11], management, environmental monitoring, industrial [12], and home automation [13]. A sensor-equipped car can monitor traffic anywhere in the city due to a camera feeding on the road, and it uses IoT sensor fusion techniques to transmit this information as feedback to the user [14,15]. Sensor fusion techniques increase the efficiency of these applications and enable the development of more intelligent and predictive systems.

Data security and privacy from IoT devices are critical for the effectiveness and reliability of sensor fusion techniques. Because these techniques aim to achieve more robust and comprehensive results by integrating these data. In this context, Ding et al. [16] classified IoT applications in various fields and proposed data integration requirements regarding the security and privacy of IoT data.

Various sensors, such as a Global Positioning System (GPS), LiDAR, and ultrasound, are used in automotive and navigation applications, where sensor fusion techniques provide effective solutions. For an autonomous driving task, data from these sensors can be combined to provide a complete view of the driving condition. While LiDAR sensors offer better coverage, they do not provide velocity information, and RADAR provides accurate velocity data but is ineffective on winding lanes. In this context, sensor fusion techniques are applied to avoid collisions, mainly to prevent false positive cases and to improve the detection quality [17].

The quadrotor navigation system usually has a complementary sensor group consisting of a three-axis gyroscope, a three-axis accelerometer, a magnetometer, a pressure altimeter, ultrasonic sensors, and GPS [18]. In this type of quadrotor and drone applications, sensor fusion techniques are applied to avoid compromising the operation if a sensor input is missing. Due to the sensor fusion technique, reliability and operational continuity are ensured.

Sensor fusion techniques enable the detection of environmental information with increased sensitivity and accuracy in computer vision applications. A computer vision study that emulates human vision using competing sensors has demonstrated this capability [19]. Additionally, sensor fusion techniques are needed in situations such as combining infrared images and multiple images with different exposures [20] or automatically scanning bags and belongings for security purposes in places such as stadiums and museums [21].

Accurately tracking head movements is a significant challenge in Virtual Reality (VR) applications. Virtual objects must be aligned with the natural world as users move their heads. In this case, a single gyroscope or accelerometer alone is not enough. VR systems overcome this challenge by using sensor fusion techniques that combine data from sensors such as gyroscopes and accelerometers. The gyroscope reduces noise in short-term movements, while the accelerometer provides long-term stability [22].

Sensor fusion techniques used in healthcare can be used for applications such as monitoring the health status of older adults and evaluating body postures in babies [23,24]. These applications use body and wireless sensor networks to fusion data from various sensors for human tracking [25], identification [26], and monitoring patients' mental states [27]. With sensor fusion techniques, the reliability of measurements is increased and false favourable rates are reduced.

Sensor fusion techniques are complex processes involving integrating data from multiple sources. Therefore, although they provide advantages and convenience in complex applications, they include challenges in implementation [28]. These challenges can be listed as 1) different operating principles of different sensors, 2) data inconsistency or errors, 3) time synchronisation, 4) high computational power requirement, 5) accuracy and reliability. Sensors with different operating principles produce various types of data. Harmonising and combining this data is a significant challenge. Although inherently sensitive, sensors can sometimes cause data loss and erroneous or noisy data. This situation is a considerable challenge to obtaining accurate results. Effective data integration from various sensors necessitates time synchronisation—a lack of synchronisation results in data being combined or misinterpreted. Sensor fusion techniques require high computational power, especially in real-time applications. This situation is a significant challenge regarding processor power or memory usage. Most sensors generate a signal through several transformation steps. Therefore, the user's output may differ from the actual input. These performance-related parameters or specifications indicate rates of deviation from ideal behaviour. While static properties such as accuracy, precision, resolution, and sensitivity can be easily managed before the fusion process, dynamic properties vary between different inputs. Therefore, there may be deviations and errors in the required information. Errors are sometimes caused by random noise and sometimes by a systematic error related to time. If the error is known, it can be fixed by a defined filter. As a result, obtaining precise and reliable results due to sensor fusion is a considerable challenge. Appropriate sensor fusion techniques must be used to overcome this challenge, and the system must be constantly calibrated. The existing literature's taxonomy is often limited to specific application areas or sensor types, making it difficult for researchers and engineers to obtain the comprehensive perspective they need. This study presents a comprehensive taxonomy of sensor fusion techniques used in complex applications, aiming to help researchers who want to work in this field choose the appropriate technique for their needs. The contributions of the study are as follows:

- The study discusses the importance of sensor fusion, its working principles, different techniques, advantages, challenges, and application areas.
- A comprehensive taxonomy of sensor fusion techniques is presented. This taxonomy allows the classification of various sensor fusion techniques, providing researchers with a comprehensive perspective on choosing the appropriate method.
- The study examines the algorithms used in sensor fusion techniques available in the literature and presents these algorithm's descriptions and mathematical formulas.

The second chapter of the study presents the general taxonomy of sensor fusion techniques. The third section presents current algorithms used in sensor fusion techniques and their application areas. The fourth section discusses the current status of sensor fusion techniques, the reasons for their preference, and offers suggestions for future sensor fusion studies.

2 Sensor fusion taxonomy

There are multiple approaches in the literature regarding the taxonomy of sensor fusion techniques. In this section, existing sensor fusion techniques are examined in detail, and a comprehensive taxonomy is created. This taxonomy makes it possible to evaluate a wide range of sensor fusion techniques and categorise them to appeal to a broader application area. Elmenreich [29] has defined three basic ways of combining sensor data: competitive, complementary, and cooperative. In another study, sensor fusion techniques have been discussed in four primary categories [30]. These categories are classification based on the relationship between different input sources, classification based on input and output data types, classification based on the abstraction level of combined data, and classification based on the kind of fusion architecture. Another study discussed sensor fusion architectures at the decision, feature extraction, and raw data levels [31]. A survey that performed data fusion taxonomy for Wireless Sensor NetworksNetworks (WSN) classified it as the relationship among the sources, levels of abstraction, input and output, data level fusion, data type fusion, and data fusion based on user requirements [32]. In addition, recent studies have categorised sensor fusion techniques as early fusion, mid/halfway fusion, and late/decision fusion [33-35].

This study examined sensor fusion taxonomies in the literature and created a new comprehensive taxonomy containing each category. The created sensor fusion taxonomy is in Figure 1.

2.1 Relationship among the sources

The relationships among the sources are categorised as “complementary”, “redundant/competitive”, and “cooperative” as in Figure 2 [36].

Complementary fusion combines data from multiple sensor nodes to obtain more general information [37,38]. In an area where one resource is missing or weak, another strong resource complements the weak or missing one. Integrating lidar and radar sensors in autonomous vehicles is a good example of this technique [39]. Camera sensors provide image-based information and detect the colours, shapes, and details of surrounding objects with high resolution. However, its performance decreases in low light conditions or adverse weather conditions such as dense fog. Lidar sensors use laser beams to provide precise distance and shape information and can create detailed 3D maps of the environment. However, certain weather conditions, like heavy rain or snow, can affect lidar performance. When these two types of sensors are combined (complementary fusion), more comprehensive and reliable information about the vehicle's environment is obtained.

Redundant/competitive fusion: Data is combined to obtain high-quality information and eliminate unnecessary data transfer. Resources are interchangeable to achieve the same purpose or goal. There are situations where one resource becomes redundant or competes with another. GPS and Lidar sensors used in autonomous vehicles and location and environmental sensing systems are good examples of redundant/competitive fusion [40]. GPS determines the vehicle's location using signals from satellites. Lidar uses laser beams to measure the distance of surrounding objects and creates a three-dimensional map. When both systems operate simultaneously, GPS and Lidar data are constantly compared to determine which sensor provides the most reliable and accurate information about the vehicle's location and surroundings. The system determines which sensor provides more reliable information and ignores data from the other sensor when necessary.

Cooperative fusion: Data from independent sources are combined to obtain new information using angle and distance. This method aims to get a more comprehensive or accurate result by combining data from different sources. Data from different sources provide different perspectives or information about the same event or target, resulting in higher accuracy and more comprehensive determinations. Detecting and tracking pedestrians in autonomous vehicles is critical for safe driving [41]. This system applies collaborative fusion to obtain more accurate and reliable results using various sensor types. Cameras provide visual information about objects (including pedestrians) around the vehicle by capturing high-resolution video or photos. A radar sensor is not particularly affected by weather conditions and can accurately detect the velocity of moving objects. Lidar measures the distance and position of surrounding objects millimeter precision and provides a detailed 3D map of the area around those objects. Visual information from the

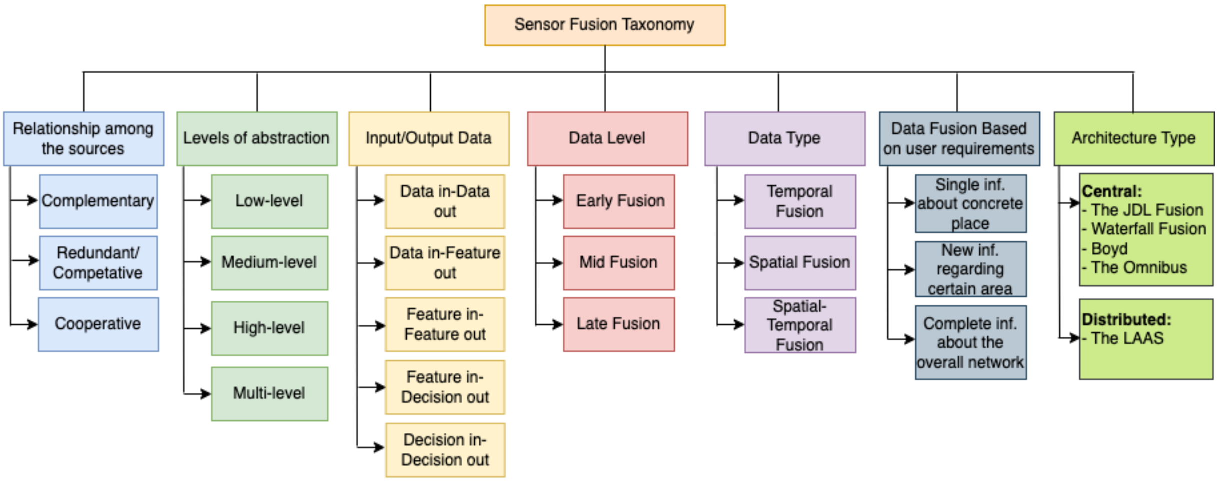

Figure 1. Sensor fusion taxonomy.

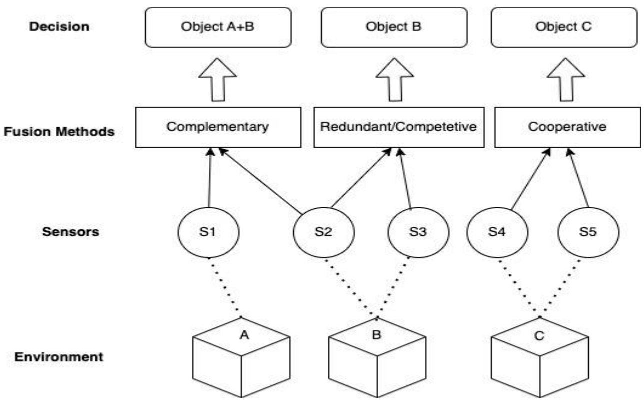

Figure 2. Complementary, redundant/competitive, cooperative fusion [29].

camera is combined with distance and velocity information from Radar and Lidar. This provides more comprehensive and accurate pedestrian detection by leveraging the strengths of each sensor. Due to this collaborative fusion, the system accurately determines pedestrians' location, distance, and velocity and monitors the environment around them in detail.

2.2 Levels of abstraction

Levels of abstraction are categorised as low-level, medium-level, high-level and multi-level [42].

Low-level fusion: Raw data is combined to reduce noise and obtain more accurate data. Each sensor alone may be incomplete or susceptible to noise. For example, in a navigation device, the magnetometer determines the general direction, while the gyroscope's data corrects the magnetometer's deviations and increases the device's stability.

Medium-level fusion: This is also called feature fusion. Features of the raw data are combined to create a feature map. Feature fusion aims to create a more comprehensive feature map by combining features of raw data from different sensors or data sources. For example, in an image processing application, a 3D feature map of an object can be created by combining the colour, depth and thermal properties of images taken from cameras operating at different frequencies. This map is used to describe and understand various features of the object in more detail.

High-level fusion: High-level fusion, also known as decision-level fusion, involves aggregating decisions as input and amalgamating them to yield more comprehensive decisions at a global scale. For example, a security system uses facial and fingerprint recognition to verify a user's identity [43]. Analyzing the user's face generates an authentication score. Scanning the user's fingerprint generates an individual authentication score. Both systems attempt to authenticate the user independently. The two resulting verification scores (facial recognition and fingerprint) are combined (for example, using a weighted average or rules based on the combined scores). The final decision determines whether the user will be granted access. In this process, high-level fusion combines the independent decisions made by each sensor to create a more reliable authentication decision.

Multi-level fusion: In this fusion level, the input and output of the data fusion system are among the previous levels. It is when the features combined at the last level are used as input to the decision-making process at the next level. For example, a smart farming system collects data about a field using various sensors and uses this data to optimize plant health and productivity [44]. In multi-level fusion, raw data from soil moisture sensors and data from weather sensors are combined at a low level. These combined data are analyzed at the intermediate level together with plant health characteristics (e.g. NDVI from drone imagery). The results of medium-level analyses are used as input to high-level decision-making. In this process, integrating data at various levels helps the farmer make correct irrigation and fertilization decisions.

2.3 Input/Output data

There are five categories of data fusion based on data input and output [45].

Data in, data out (DAI-DAO): Within this category, the data fusion system receives raw data as input and produces more dependable raw data as output.

Data in, feature out (DAI-FEO): It involves feeding raw data into the data fusion system, yielding an extracted property or attribute, such as an object or state.

Feature in, feature out (FEI-FEO): Here, the system takes a feature as input, generates a verified feature, or extracts new features.

Feature in, decision out (FEI-DEO): In this category, a set of features serves as input to the data fusion process, resulting in decision-making outputs.

Decision in, decision out (DEI-DEO): This category encompasses scenarios in which decisions are provided as input and new decisions are generated as output through the data fusion process.

2.4 Data level

Sensor data can be combined at different levels in different applications. Three different levels of data fusion are shown in Figure 3.

Early fusion: Features from all sensors are combined into one input, and one classifier is trained on the combined feature representations. It integrates information from different methods at the feature level and requires only one classifier. It can be computationally efficient. However, it carries the risk of high input dimensionality, which can lead to overfitting and increased implementation costs. It may exhibit unusual or non-intuitive properties due to the combination of various types of information.

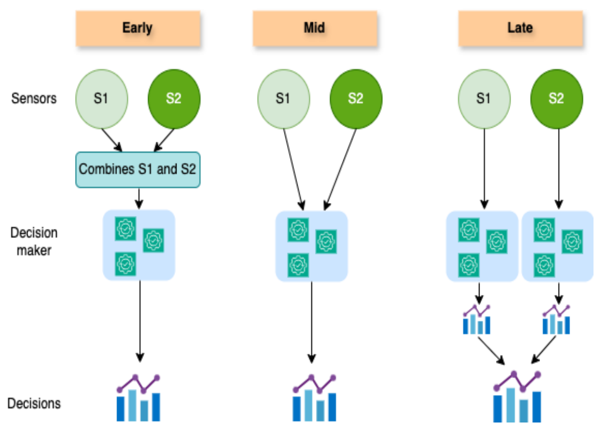

Figure 3. Data level fusion techniques [46].

Mid fusion: There is one classifier, but it is trained on a properly processed, more abstract version of the input from each modality. Extracting abstract representations from each modality requires additional processing and can be computationally costly.

Late fusion: Each modality is trained with a separate classifier that makes decisions independently, and then the decisions or predictions from the individual classifiers are combined to obtain the final classification result. This fusion allows each modality to be processed independently and can be computationally efficient. It can handle unit and scale differences between modalities. However, it involves the risk of missing complementary information in the combined feature representation. Additional processing is required to integrate the outputs of individual classifiers.

2.5 Data type

Depending on the data type, there are three types of data integration: "temporal fusion, spatial fusion, and temporal-spatial fusion" [47].

Temporal fusion means combining data from different periods but from the same source. Thus, time series data can detect changes and distortions in time and capture seasonality and trends.

Spatial fusion means combining data simultaneously but from different sources. Thus, the same event or phenomenon can be examined from various perspectives. It provides other features and offers multiple perspectives for an event identified by different sensors (such as local and satellite).

Temporal spatial fusion refers to continuously merging data from different nodes over a certain period. It is suitable for studies that require continuous data merging to obtain an instantaneous status. It can capture both changes over time and relationships in different places.

These methods can meet different data analysis and processing needs and are often used to transform extensive data sets into more meaningful and usable information.

2.6 Data fusion based on user requirements

Depending on the user's needs, there are three types of data fusion: single, new, and complete. Sometimes, the user may need a *single* piece of information about a concrete place that can be obtained with a *single* sensor or *new* information about a specific area. In addition, the user may need *complete* information about the general location or network [48]. For example, the user wants to monitor room temperature changes using data from a single sensor. In this case, the user only needs a *single* data set for a specific point. In cases where the user wants to create a new usage map by combining data obtained from satellite images and local sensors, the aim is to get *new* information by combining existing data. In another case, an urban planner may want to analyse traffic flow throughout the city and optimise the transportation system. In this case, the user aims to create a *complete* map of the city-wide transportation network by combining data from different sources.

2.7 Architecture type

This study discusses sensor fusion architecture types in two categories: central and distributed. *Central architecture* is the approach in which data from different sources is combined in a central location. The fusion process is performed on a central server or computer. *Distributed fusion* architecture is an approach in which data from different sources is combined in a distributed manner. In this approach, the fusion process is performed distributedly where resources are available.

2.7.1 Central architecture

Central fusion architecture collects data fusion and processing processes at a single central point, as in Figure 4. This architecture suits situations where data flows are collected and managed under central control. It provides data consolidation and captures an integrated perspective. Central architecture is advantageous because it is simple and optimal. It also allows for the easy detection of erroneous reports [49].

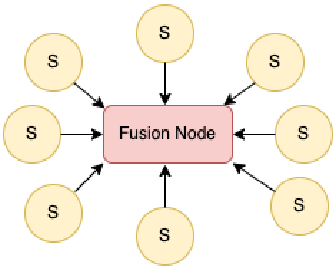

Figure 4. Central architecture.

On the other hand, this architecture needs higher bandwidth to transmit data from all sensing nodes to the central processor fusion and requires more resources for data processing [50]. One of the central models used in both military and commercial fields is JDL. JDL model includes five levels of data processing and a database, all connected by a data path [51]. The elements of the model are as follows:

Sources include various data sources such as sensors, a priori information, databases, and human input.

Source preprocessing (Level 0): This element pre-scans data and allocates it to appropriate processes, reducing the processing load of fusion processes.

Object refinement (Level 1): This level provides data alignment (data transformation into a consistent reference frame), correlation, tracking of objects' actual and future locations, and identification using classification methods.

Situation refinement (Level 2): Situation refinement attempts to contextualise the relationship between objects and observed events.

Threat refinement (Level 3): Attempts to make inferences about vulnerabilities and operational opportunities based on preliminary information and predictions.

Process refinement (Level 4): This meta-process monitors system performance (e.g., real-time constraints) and reallocates sensors and resources to achieve specific mission objectives.

Database management system: This system monitors, evaluates, adds, updates, and provides information about fusion processes.

Man-machine interaction contains an interface that transmits input and fusion results to operators and users.

The JDL model, being data or information-centric, poses challenges in extending or repurposing applications developed under its framework. Its abstract nature complicates accurately interpreting its components and their application to specific problem domains. While the model aids in establishing a common understanding, it lacks guidance for developers in selecting appropriate methodologies. JDL fusion architecture can be used in application areas such as military [52], agriculture [53] and autonomous vehicles [54].

Another central fusion architecture is waterfall architecture, as shown in Figure 5. Waterfall emphasises processing functions at lower levels [55]. The process stages of the waterfall model consist of levels 0,1,2,3. *The detection and signal processing level (level 0)* corresponds to source preprocessing in the JDL model. *The feature extraction and pattern processing level (level 1)* matches the object refinement in the JDL model. *The situation assessment level (level 3)* matches the situation improvement level in the JDL model. *The decision level* corresponds to the threat improvement level in the JDL model. The waterfall model, which is very similar to the JDL model, has the same disadvantages. The most significant limitation of the waterfall model is that any feedback data flow is neglected. Waterfall fusion model is used in application areas such as agriculture [56], health [57] and maintenance system [58].

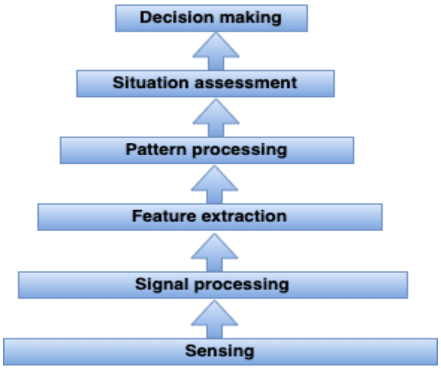

Figure 5. Waterfall model.

Another known central architecture is the Boyd model [59], which consists of a 4-stage cycle, as shown in Figure 6. The Boyd loop is used for sensor fusion because decision support systems for situational awareness are strongly related to fusion systems.

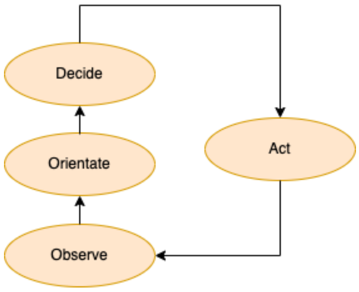

Figure 6. Boyd model.

The elements of the Boyd model are divided into four. *Observe* generally corresponds to resource preprocessing in the JDL model. Orientate corresponds to level 1, 2, and 3 functions of the JDL model. Decide is comparable to level 4 (process improvement) of the JDL model. The act includes implementing decisions. In this phase, decisions are translated and implemented into real-world actions. Actions are based on decisions taken to achieve predetermined goals. It has no direct equivalent in the JDL model. First used to model the military chain of command, the Boyd fusion model became widespread for data fusion.

Another central architecture similar to the Boyd model is the omnibus model in Figure 7 [60]. However, unlike the

Boyd model, the omnibus model structures the transaction levels in detail.

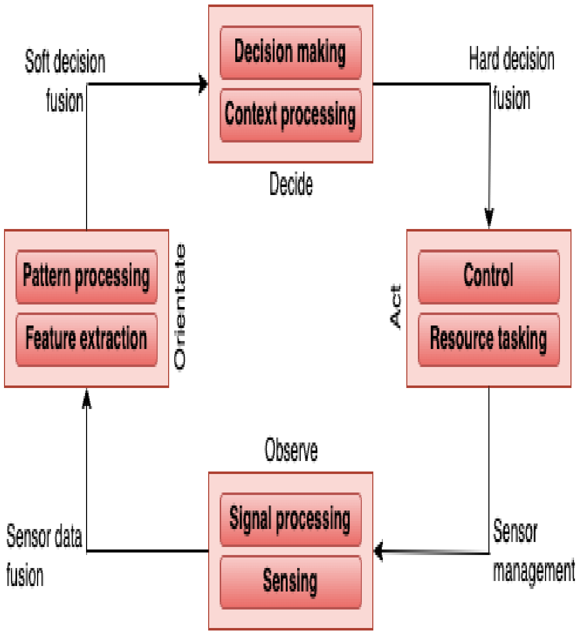

Figure 7. Omnibus model.

While the hierarchical division of sensor fusion tasks in the omnibus model is notably intricate, it cannot partition functions in a manner conducive to distributed sensing and data processing. Consequently, the model does not facilitate decomposition into modules that can be independently implemented, tested, and repurposed for diverse applications. Table 1 presents the advantages, disadvantages, and reasons for preference of the introduced central architectural fusion architectures.

2.7.2 *Distributed architecture*

Unlike a centralised architecture, a distributed architecture does not have a single central node, as in Figure 8. Nonetheless, data fusion is locally executed at every node within the network, leveraging observations obtained from adjacent nodes. Distributed architecture provides advantages in terms of supporting changes in the network, scalability and tolerance [61]. Data is sent to multiple nodes instead of a central node, reducing processing load and communication overhead. Additionally, there is less communication delay, allowing the user to access fusion results faster. One of the known distributed architectures is the Laboratoire d'Analyse et d'Architecture des Systemes (LAAS) architecture [62].

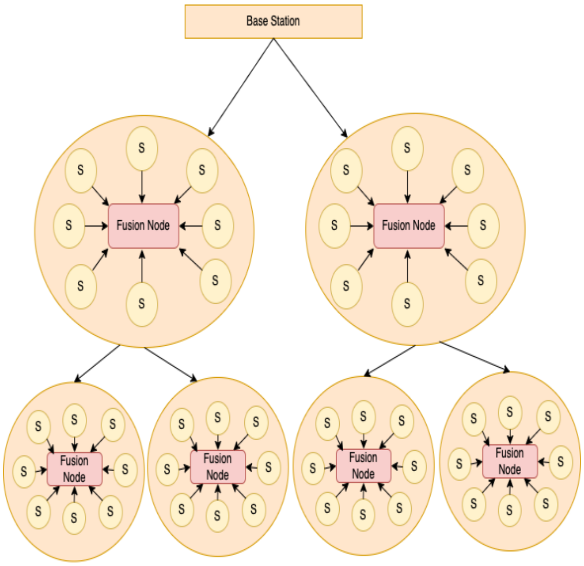

Figure 8. Distributed architecture.

Table 1. Central fusion architectures.

Model	Advantages	Disadvantages	Reason of preference
JDL	Can integrate complex data sources and processes.	High bandwidth requirements. Huge resource requirement for data processing.	It is used when integrating a wide variety of data sources is required. It has been widely accepted and used in military and commercial fields.
Waterfall	The clear and distinct separation of stages. Each stage focuses on a specific function. Simple and understandable structure.	Feedback flow is not taken into account. Lack of flexibility between stages.	It is used in applications where a simple and flat process structure is needed. It is preferred in areas such as agriculture, health and care systems.
Boyd	It is suitable for decision support systems. It provides a good structure for situational awareness. It includes a feedback mechanism between stages.	It requires high data flow and processing load under central control. It is not flexible due to its non-modular structure.	It is used in military decision-making processes. It is preferred in applications where decision and action cycles are frequent.
Omnibus	Detailed and comprehensive phase separation. Provides detailed functional structure for complex tasks. Hierarchical separation of processing levels.	Lack of separability in modular construction. High complexity and resource requirement.	It is suitable for mixed and multi-stage sensor fusion missions. It is used when modularity is unnecessary and centralised control is required.

LAAS has been developed as an integrated architecture for designing and implementing mobile robots in real time and for reusing code. LAAS architecture consists of four levels.

The logical robot level creates a hardware-independent interface between physical sensors, actuators, and the functional level.

Processing functions such as *image processing level*, obstacle avoidance, and control loops are housed in separate controllable communication modules.

Execution control level controls and coordinates the execution of functions according to mission requirements. *The decision level* includes the ability to generate the mission plan and oversee its execution, as well as react to other events at the executive control level. Depending on the application, the decision level may consist of several layers that provide different representation abstractions and have different temporal properties. While the architecture provides a suitable means of dividing large systems into modules, it does not offer real-time communication and representation of aggregated data. Centralized and distributed fusion architectures are compared in Table 2.

Table 2. Central and distributed fusion architecture comparison.

Feature	Central	Distributed
Structure	Data fusion and processing is done at one central point.	Data fusion is performed locally at each node in the network.
Models	JDL, Waterfall, Boyd, Omnibus	LAAS
Advantages	It is the data management and control center. Errors can be easily detected with centralized control. Provides data integration and consolidation.	Provides support for changes in the network. It offers scalability and tolerance. Processing and communication load is reduced. Faster access and less communication delay.
Disadvantages	High bandwidth and data processing requirements. Failure of the central node affects the entire system. Lack of modular structure.	Complex management and coordination due to distributed structure. Lack of unified data representation and real-time communication.
Scalability	Low	High
Reaction time	High	Low
Data Processing Load	High	Low
Communication Payload	High	Low
Modularity and Reusability	Medium	High
Ease of Usage	High	Complex

3 Algorithms used in sensor fusion applications

Sensor fusion combines data from multiple sensors to obtain more consistent, reliable and accurate information. This technique is used in various applications and increases system performance due to the different features and capabilities of the algorithms. With developing technologies, a wide range of sensor fusion algorithms are used, from classical methods such as Kalman Filters (EKF and UKF) to modern techniques such as Deep Learning (DNN, CNN, LSTM) and Graph Neural Network (GNN). These algorithms significantly contribute by providing effective solutions in various fields, such as autonomous driving, robotic control, healthcare and rehabilitation. Each of these different algorithms can be optimized according to application requirements and tailored to perform best on specific tasks. For example, in scenarios such as home rehabilitation studies, algorithms such as Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) have been used to estimate motion by integrating inertial and visual sensor data [63]. Studies have shown that both fusion algorithms offer similar accuracy levels, but UKF has higher computational power.

Table 3. The reviewed sensor fusion applications according to their algorithms and categories.

Reference	Category	Algorithm
[63]	Probabilistic	EKF and UKF
[64]	DL-Based and Probabilistic	EKF, PF, LSTM
[65]	DL-Based	DNN
[66]	DL-Based	TransFuser
[67]	DL-Based	CNN
[68]	DL-Based	AutoEncoder
[69]	DL-Based	Transformer
[70]	DL-Based	MFIN
[71]	DL-Based	RNN
[72]	DL-Based	LSTM
[73]	ML-Based	AdaBoost
[75]	Probabilistic	Dempster Shafer
[78]	DL-Based	RNN
[79]	DL-Based	CNN
[80,81]	DL-Based	RCNN
[82]	DL-Based	SPP-Net
[83]	DL-Based	Fast-RCNN
[84,85]	DL-Based	Faster-RCNN
[87,88]	DL-Based	YOLO
[94,95]	DL-Based	SSD
[97,98]	DL-Based	DSSD
[99]	Feature Maps	Network Scan

Information from force, encoder, and visual detection is integrated with differentiable filters (EKF, Particle Filter (PF), LSTM) to perform state estimation in specified tasks. The results have been compared with Bayesian filters and crossmodal weighted fusion methods [64].

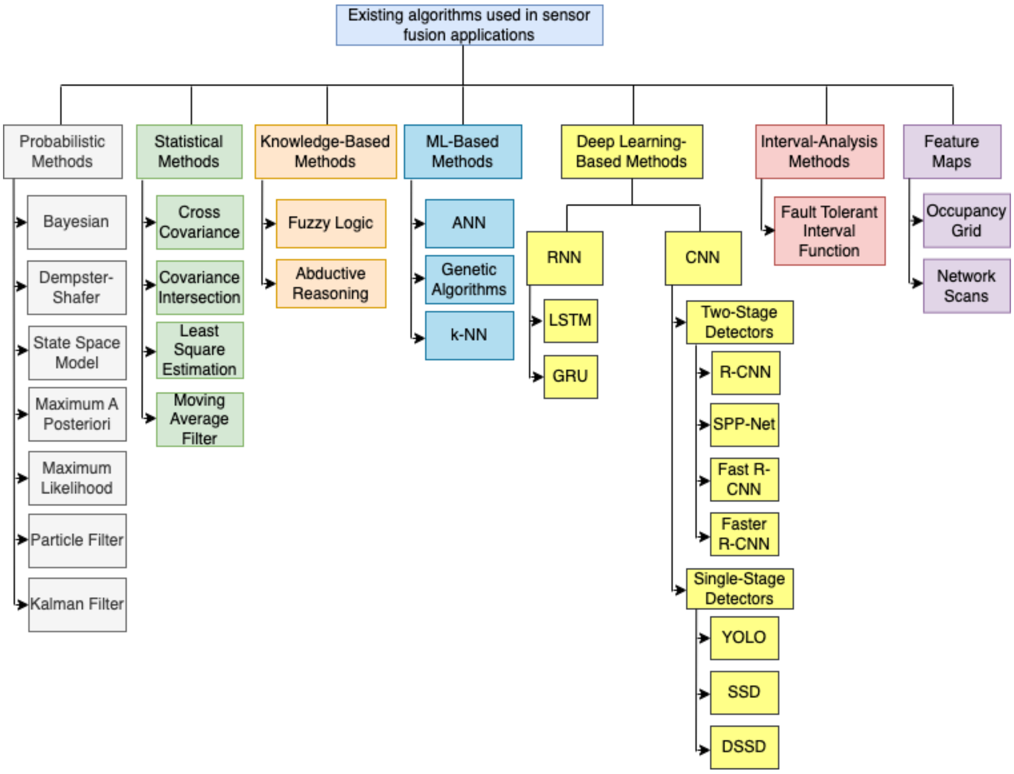

Figure 9. The algorithms used in sensor fusion applications in the literature.

On the other hand, Deep Learning and multi-modal sensor fusion techniques have been used to increase autonomous driving performance and generalization ability. For example, RGBD and Lidar sensors have been integrated using the Deep Neural Network (DNN) algorithm and 100% success has been achieved in training and static navigation tasks [65]. The TransFuser method, on the other hand, integrates image and Lidar data for finite driving using self-attention [66]. This method uses multiple transducer modules to fuse perspective views and feature maps, reducing average collisions per kilometer by 48% compared to geometric fusion. RGB and Lidar sensor data are combined with CNN to optimize environmental sensing of autonomous driving systems in complex urban environments [67]. Another study using sensor fusion techniques for autonomous driving systems introduced a new method called BEVFusion [68]. Using multi-sensor data, BEVFusion combines LiDAR and camera features in a bird's-eye view (BEV) representation space while preserving geometric and semantic information. BEVFusion transforms multi-modality features like LiDAR and camera into a unified bird's eye view (BEV) representation space with efficient view-to-view transformations. It processes these combined BEV features with a fully convolutional BEV autoencoder and supports different 3D sensing tasks with task-based heads.

In another study, the Transformer-based sensor fusion method, which can be used in many sensing systems, such as autonomous driving and robotics, has been examined [69].

Integrating body sensor networks (BSN) in medical services has gained significant importance. One study used an interpretable neural network (MFIN) method for BSN integration by combining various sensor, communication, robotic, and data processing technologies [70]. MFIN uses CNN, Recursive Neural Network (RNN), CapsNet and Random Forest (RF) algorithms to extract features from sensor data, and Graph Neural Network (GNN) and CapsNet algorithms to correlate features. The obtained features are combined with a Bayesian network. In another study, RNN has been used for BSNs integration [71]. In another study performing Human Activity Recognition, multimodal sensor fusion consisting of IMU sensors and measurements of an Android device has been performed with the LSTM model [72].

Additionally, in complex tasks such as disease prediction, the AdaBoost machine learning algorithm has been used to integrate various sensor data such as pressure, light sensitivity, motion, electrical signal, current, acceleration, and angular velocity [73]. In this review, sensor fusion techniques and sensor fusion algorithms are categorised separately. There are multiple classifications of sensor fusion algorithms in the literature. Elmenreich [29] classified some sensor fusion algorithms as state estimation and decision fusion methods. In [30], sensor fusion algorithms are classified in two different ways: classical fusion algorithms and deep learning fusion algorithms. Sensor fusion algorithms for WSNs are divided into categories: inference, estimation, compression, reliable abstract sensors, feature map, aggregation and an information theory approach [32]. In addition, recent studies used DL-based algorithms in sensor fusion applications [33-35]. In this section, the existing algorithms used in sensor fusion applications in the literature are categorised as in Figure 9. Additionally, Table 3 summarises the reviewed sensor fusion applications according to their algorithms and categories.

3.1 Probabilistic methods

Probabilistic methods estimate probability distributions of various situations to make accurate and reliable predictions under uncertainty. It is more common to use it in uncertain and complex systems. Existing probabilistic method algorithms are as follows:

Bayesian: The Bayesian approach provides a theoretical framework that deals with uncertainties utilising a foundational graphical framework. This approach is well-suited for analysing past events and forecasting the probability of various causes contributing to their occurrence. Mathematically, it is as in Equation (1) [74].

$$P(C|D) = \frac{P(C|D)P(C)}{P(D)} \tag{1}$$

In this context, P(C) represents the likelihood of event C occurring independently, while P(D) signifies the probability of event D unfolding without any external influence. Meanwhile, P(D|C) denotes the probability of event D happening, given that event C has already transpired. The resulting probability of the condition, P(C|D), falls within the range of zero to one [1 0], indicating the likelihood of event C occurring given event D. In other words, this signifies event P(C|D). Although Bayesian theory has the advantage of simplicity of calculation and high probabilities for the correct decision, it has some disadvantages. These disadvantages include difficulty recognising decision uncertainty, complexity due to many events that depend on multiple hypotheses and conditions, and difficulty determining the value of prior probabilities.

Dempster-Shafer: The Dempster-Shafer method is a probability theory-based approach that enables the combination of different evidence under uncertainty and the making of predictions by considering situations where this evidence may conflict with each other. It has become popular in applications such as signal decoding and recognition and pattern recognition. It provides a vital formula that combines a variety of evidence from different sources. The Dempster Shafer has been evaluated for various perceptual tasks, encompassing sensor fusion, scene interpretation, recognition of object targets, and object validation. The method combines information from different sources. It uses belief and probability values to represent evidence and corresponding uncertainty [75,76]. The technique uses "belief" instead of probability. The Belief function is used to describe the uncertainty of the hypothesis. The hypothesis probability is defined by the mass function m. The amount of belief in a hypothesis (A) is represented by a belief function [77].

$$Bel(A) = \sum_{B \subseteq A} m(B) \tag{2}$$

In Equation (2), the sum of the mass probabilities of all subsets of A assigned according to m is calculated. The presence of two or more pieces of evidence is combined using the join rule in Equation (3).

$$m(A) = \frac{\sum_{i,j} m_1(B_i).m_2(C_j) \quad B_i \cap C_j = A}{1 - k} \tag{3}$$

Here, 1-k is a normalisation factor, which is the sum of all non-zero values given to the ∅ hypothesis. A feature is classified based on the maximum belief decision rule, which assigns it to class A if the total amount of beliefs supporting A is more significant than those not supporting A. This situation is expressed mathematically as in Equation (4).

$$Bel(A) \geq Bel(\bar{A}) \tag{4}$$

State-Space Model: A state-Space model is a probability-based approach used for the mathematical modelling of dynamic systems. It describes the state variables in a system and how these variables evolve. The state space model consists of two primary components: the state equation and the measurement equation. The state equation describes how the state variables change over time and expresses the system dynamics as a first-order differential equation. Equation (5) shows the state equation:

$$x_{k+1} = F_k x_k + B_k \mu_k + w_k \tag{5}$$

x_{k+1} represents the value of the state in the system at the next step. x_k is the value of the system state at the current time step. F_k is the transition matrix that indicates how the system evolves. B_k is the control input matrix and isuggeststhe effect of control inputs on state inputs, if any. μ_k represents control inputs. w_k represents a continuous and discontinuous random process (usually noise). The measurement equation relates the observable outputs of the system to state variables. It indicates the relationship between real-world data and the model and is calculated as in Equation (6).

$$y_k = H_k x_k + v_k \tag{6}$$

y_k represents the measurement vectors received from the system.x_k represents the state in the system. H_k is the measurement matrix that shows how state variables in the system are associated with measurements. v_k represents measurement error or noise.

Maximum a Posteriori: This approach is a probabilistic approach that determines the most probable value of a parameter using Bayes' theorem. It estimates parameters by modelling the relationship between prior information about a parameter or event and observed data. According to Bayes' theorem, the posterior distribution of a parameter is calculated based on priori and observed data. The maximum a posteriori selects the value at the maximum point of this posterior distribution, which provides the most probable estimate of the parameter. This method is a version of maximum likelihood that considers previous information about a parameter or event. If prior information is available and reliable, the maximum a posteriori estimate may be more robust than the maximum likelihood estimate. The maximum a posteriori estimate based on Bayes' theorem is calculated as in Equation (7).

$$P(\theta|D) = \frac{P(D|\theta).P(\theta)}{P(D)} \tag{7}$$

Here, P(θ | D) represents the a posteriori probability of parameter θ when data D is observed. P(D | θ) represents the probability of parameter θ when data D is observed. P(θ) represents the prior probability of θ, and P(D) represents the marginal probability of data D. Maximum a Posteriori selects the parameter θ value at the maximum point of the a posteriori likelihood as in Equation (8).

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, P(\theta|D) \tag{8}$$

When data from multiple sensors in a system must be combined to predict or measure the system state, the measurement of each sensor may have a different probability distribution. Combining a priori information with the maximum a posteriori can provide the most accurate estimate in this situation.

Maximum Likelihood is a probabilistic method for estimating the most likely value of a parameter by combining measurements from different sensors. It considers the probability distributions of each sensor measurement and combines these distributions to provide the most probable estimate. Firstly, measurement probability distributions from each sensor are calculated. These distributions express uncertainty around the actual value of the measurement. Secondly, the calculated probability distributions are combined. A probability distribution is obtained by combining measurements from all sensors. Considering each sensor measurement, a distribution containing the most probable value of the system state is obtained. Thirdly, the combined probability distribution makes the most likely prediction of system parameters. Finally, system status estimates are updated and improved with new sensor measurements. More accurate predictions are obtained when sensor data are combined with maximum likelihood. However, this method requires obtaining accurate probability distributions of each sensor measurement and may present some practical difficulties. Let the model parameters be assumed to be θ and the observed data be D. The likelihood function for the data set indicates the probability of the data set under the parameter θ and is denoted by P(D| θ). The most probable parameter estimate is obtained by maximising the likelihood function. Mathematically possible parameter estimation is as in Equation (9).

$$\hat{\theta}_{maximum\ likelihood} = \underset{\theta}{\operatorname{argmax}}\, P(D|\theta) \tag{9}$$

Likelihood functions often involve multiplication operations, and the logarithmic functions convert multiplication operations to addition operations. This makes the calculation easier. Maximum likelihood estimates are maximised with the log-likelihood function and calculated as in Equation (10).

$$\hat{\theta}_{maximum\ likelihood} = \underset{\theta}{\operatorname{argmax}}\, \log P(D|\theta) \tag{10}$$

Particle Filter: The PF method uses a Sequential Monte Carlo (SMC) to solve the state estimation problem and can approximate Probability Density Functions (PDFs). PF involves a resampling step at each instant, using the Sequential Importance Sampling (SIS) algorithm. The density function is created by several random samples called particles. In the particle production stage, initial particles p(x(0)) are generated according to the initial probability density function (PDF) N{x1(0), x2(0), x3(0), . . . , xN(0)}. In the estimation phase, each particle x_i (k + 1) is propagated according to p(x(k + 1) | x(k)), which is the PDF of x_i (k + 1). Each particle calculates the sum of random noise to simulate the noise effect. In the sampling phase, w_i (k + 1) = p[z(k + 1) | x_i (k + 1)] is created for each x_i (k + 1) particle. In the normalisation and rejected samples stage, the weights of the particles are normalised. Low-weight particles are removed, and high-weight particles are copied so that each particle has the same weight. PF is evaluated in traffic control, military field, mobile robot positioning and self-positioning.

Kalman Filter (KF): The Kalman Filter (KF) algorithm estimates the state of a discrete time-controlled process characterised by a linear stochastic equation. It relates the state from the previous time step to the current measurement to accurately extract the current state. This method is a preferred algorithm, mainly due to its direct applicability to linear systems. Kalman Filter formulation is as in Equation (11).

$$\begin{aligned} \widehat{\mathrm{x_k}} &= \widehat{\mathrm{F_k x_{k-1} + B_k \mu_k}} \\ P_k &= F_k P_{k-1} F_k^T + Q_k \end{aligned} \tag{11}$$

$\widehat{\mathrm{x_k}}$ is state vector of $\mathrm{x_k}$ system. P_k is the estimated covariance matrix. F is the dynamics of the system matrix. B refers to the control matrix, and Q refers to the noise covariance. KF is used to generate new predictions by adding an external unit for correction. The KF includes the step in Equation (12) to update the state.

$$\begin{aligned} \widehat{x_k'} &= \widehat{x_k} + K'(z_k - H_k \widehat{x_k}) \\ P_k' &= P_k - k' H_k P_k \\ K' &= P_k H_k^T (H_k P_k H_k^T + R_k)^{-1} \end{aligned} \tag{12}$$

Here, z_k contains the measurement vectors from the sensors. H is the transformation matrix. R represents the covariance matrix of noise measurements and the k time interval. Kalman gain (K) refers the amount of update needed based on the relationship between prediction accuracy and measurement noise. Weighting is done on each iterative prediction. KF is the most appropriate estimator that can be used to predict the statistical behavior of measured values. In most real problems, systems cannot provide linear characteristics. EKF is suitable for this problem. The main benefit of the Kalman filter is its computational cost, but it can only express single modality distributions. Another version of the KF is the UKF.

3.2 Statistical methods

Statistical methods use statistical methods to extract meaningful information from noisy sensor measurements. Statistical method algorithms are as follows:

Cross Covariance: It is a statistical method used to measure the relationship between two variables. Cross-covariance helps determine the direction and strength of the relationship between two variables and how lagged it is. Firstly, the average of two different variables is calculated. This means calculating the average of the observed values of each variable. Secondly, the differences between the averages of both variables are taken. These differences are then multiplied and averaged. The covariance of two variables is calculated as in Equation (13).

$$Cov_{XY}(k) = \frac{1}{n}\sum_{t=1}^{n-k}(X(t+k)-\bar{X})(Y(t)-\bar{Y}) \tag{13}$$

k refers to the delay between two variables. n refers to the total number of observations. X(t) and Y(t) refer to the values of the variables X and Y at time t, respectively. $\bar{X}$ and $\bar{Y}$ refer to the averages of the variables X and Y, respectively. If the cross-covariance is positive, the relationship between the variables is positive. If the cross-covariance is negative, the relationship between the variables is negative. Additionally, the size of the cross-covariance value indicates the strength of the relationship.

Covariance Intersection: It is a statistical method that combines the covariance matrices of data from different sensors. These covariance matrices represent the uncertainty and accuracy of each sensor measurement. The method aims to obtain a more reliable estimate by combining these uncertainties. Firstly, the covariance matrices of the measurements from each sensor are calculated. Then, the calculated covariance matrices are joined to obtain a single covariance matrix. Different methods can be used during the join phase. However, the weighted join method is often used in covariance intersection. Weights are assigned based on the reliability of sensor measurements. Covariance intersection is calculated as in Equation (14).

$$\Sigma_{covarience\ intersection} = \left(\sum_{i=1}^{n}\alpha_i\Sigma_i^{-1}\right)^{-1} \tag{14}$$

Σ_i^{-1} is the inverse of the covariance matrix of each sensor, and α_i is the weight of each sensor.

Least Square Estimation aims to minimise the error between data points. It is frequently used in applications such as linear regression and curve fitting. Least square estimation tries to minimize the sum of the squares of the differences between the data points and the values predicted by the model. It is a practical and easy-to-calculate method. However, it may give misleading results in the presence of outliers and cases that do not comply with the linearity of the model. Therefore, the characteristics of the data set and the suitability of the model should be carefully evaluated. Data from sensors can often have different accuracy, sensitivity, and noise levels. Therefore, least square estimation plays an important role in obtaining accurate and reliable results by combining sensor data. The Least state estimation method increases the accuracy and reliability of sensor data and can combine different types of sensors and measurement data. It is calculated as in Equation (15).

$$\hat{\theta} = (X^T X)^{-1} X^T y \tag{15}$$

$\hat{\theta}$ represents the estimated parameter vector, X represents the design matrix containing independent variables, y represents the dependent variable vector consisting of actual values or observed results.

Moving Average Filter is a widely used statistical method for correcting or smoothing data in time series. This filter can be used to reduce random noise in the data or identify trends in the data set. The basic principle of the moving average filter is to replace each data in a time series with the average of data taken over a certain period (for example, the last N measurements). The average of measurements for a specified period provides a more accurate estimate of the original data. Firstly, a window size is selected for the filter. Secondly, the window is moved along the time series, and for each window position, the average of the measurements across the window is calculated. Finally, the calculated average value changes the relevant point of the original time series data to filtered data. In this method, choosing the correct filter size is very important. Because filter size directly affects the results. Moving average filter is calculated as in Equation (16).

$$Moving\ Average_t = \frac{1}{N}\sum_{i=0}^{N-1} x_{t-i} \tag{16}$$

$Moving\ Average_t$ is the moving average value at time t. x_{t-i} is the measurement value at time t-i. N refers to the window size.

3.3 Knowledge-based methods

Knowledge-based methods rely on domain-specific knowledge or rules. This method is used to interpret sensor measurements and make decisions. Fuzzy logic and abductive reasoning are knowledge-based sensor fusion methods.

Fuzzy Logic: This algorithm uses imprecise concepts and expressions to deal with uncertainty. Fuzzy logic uses fuzzy or gradient categories instead of strict boundaries. Therefore, it is suitable for modeling and controlling systems with uncertain or fuzzy information. For example, fuzzy logic can be used to describe quantities such as temperature, velocity or position. The fuzzy logic model is constructed by leveraging the expertise and insights of domain experts. In this method, combining is performed by considering the degrees of uncertainty and uncertainties instead of assigning sensor data to precise categories.

Abductive Reasoning generates a set of possible explanations or hypotheses and uses available evidence to determine their accuracy. This process can consider existing evidence, as well as experience, understanding, and intuition. Abductive reasoning can be used to interpret or make sense of sensor data before sensor fusion.

3.4 ML-based methods

Machine Learning-Based Methods are an important approach to integrating different sensor data and obtaining more accurate results. This method algorithms are as follows:

Artificial Neural Network (ANN): This machine-learning model draws inspiration from the structure and function of biological neural networks. In sensor fusion, ANN helps obtain more comprehensive information by processing data from multiple sensors and learning the relationship between these data. ANN is as in Equation (17).

$$y = \sigma(W_L.\sigma(W_{L-1}.\dots.\sigma(W_1.x + b_1) + b_{L-1}) + b_L) \tag{17}$$

X is the vector representing the input data and, y represents the vector of output data. W_i is the weight matrix in the i_{th} layer. b_i represents the bias vector in the i_{th} layer and σ represents the activation function. The use of ANN in a specific sensor fusion problem varies depending on the architecture of the network and the problem. It is important to adjust the network architecture and activation function depending on the application and problem.

Genetic Algorithms is an ML algorithm that mimics the principles of natural selection and genetic inheritance. It can be used in sensor fusion to find and optimize relationships between different sensor data. It looks for many combinations of traits of individuals within a population. It achieves these combinations by iteratively crossing individuals, mutating them, and selecting the best-performing individuals. In the first step, a random initial population $P_0 = \{I_1, I_2, \dots, I_N\}$ is created. This population consists of individuals representing data from sensors. In the second step, a fitness function $f(I_i)$ is created that evaluates the fitness of each individual. This function measures success in achieving a specific goal or optimizing a goal. In the third step, some of the individuals in the population are selected according to the fitness function and used for the next generation. Then, a crossover is performed between the selected individuals. This process creates new individuals by selecting the characteristics of the parents. Finally, the mutation is applied to the created individuals. The genetic structure of these individuals is randomly changed to increase diversity. As a result of crossover and mutation, a new generation is created.

k-Near-Neighbors (k-NN): Sensor fusion leverages k-Nearest Neighbors (k-NN) to cluster and merge data originating from disparate sensors. k-NN calculates the distance between two samples with distance measures such as Euclidean or Manhattan. Grouping is done by calculating the distances of each sample from all other samples and selecting the k closest samples. During grouping, a majority vote or an average value calculation can be performed between class labels. For K-NN, the distance is as in Equation (18).

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2} \tag{18}$$

3.5 Deep learning-based methods

Deep learning techniques have become popular for sensor fusion. Deep learning models can learn to extract features and combine sensor data end-to-end.

Recursive Neural Network (RNN): RNNs feed back the outputs from previous steps to the inputs from subsequent steps. This is ideal for modelling dependencies in time-series data. Distributed sensors are employed for monitoring both the vital signs and behavioural patterns of individuals. Sensor fusion applications have increased to provide valuable lifestyle-related data and recognition of physical human action through body sensors. In [78], a body sensor-based sensor fusion method has been proposed for behavior recognition using RNN. In the study, multiple body sensor fusion such as electrocardiography (ECG), accelerometer and magnetometer has been performed. RNN is as in Equation (19).

$$\begin{gathered} h_t = \sigma(W_{ih}x_t + W_{hh}h_{t-1} + b_h) \\ y_t = \sigma(W_{ho}h_t + b_0) \end{gathered} \tag{19}$$

x_t represents the input vector at time t. h_t is the hidden layer vector (RNN memory) at time t. y_t represents the output vector at time t. W_{ih} refers the weight matrix from the input layer to the hidden layer while W_{hh} refers the weight matrix from hidden layer to hidden layer. W_{ho} refers to the weight matrix from the hidden layer to the output. b_h and b_0 represent the hidden layer and output bias vectors. σ refers to activation functions such as sigmoid and tanh.

Long-short-term memory (LSTM) and Gated Recurrent Unit (GRU) models derived from the RNN architecture are specifically engineered to capture long-term dependencies within data sequences more effectively. A drawback of RNNs is that they need to capture long-term dependencies better. This is especially important for tasks where long-term dependencies are important, such as time series data. LSTM and GRU have special memory mechanisms to better model long-term dependencies. LSTM includes a cell state and three gates (input, forget, output). These gates control the insertion of information into the cell state. GRU has a simpler structure than LSTM. It consists of only two gates (reset and update). These gates control forgetting past information and adding new information. LSTM has more parameters than GRU. Therefore, LSTM is more complex and flexible than GRU. GRU training time is faster than LSTM and requires less computation.

Convolutional Neural Networks (CNN): CNNs are popular for combining or analyzing data obtained from image-based sensors. In autonomous vehicles, CNNs can be used to process images obtained from camera sensors and detect objects. However, the direct transfer of these modalities is a significant challenge due to processing multimodal sensor data and the lack of large labelled datasets. In [79], CNN has been used in multi-modal sensor fusion for human action recognition. CNN has been used to extract higher abstraction features from each sensor data. Then, these features have been combined. In the convolutional layer, filters (kernels) are applied to the input data and feature maps are produced. This process is as in Equation (20).

$$z^{[l]} = \sum_{i=1}^{n} W^{[l]} * x^{[l-1]} + b^{l} \tag{20}$$

$z^{[l]}$ is convolution layer output, $W^{[l]}$ is the filter weight matrices in the layer, $x^{[l-1]}$ is the input data from the previous layer, b^{l} is the bias vector of the layer.

The pooling layer's primary function is to diminish the dimensions of feature maps acquired post-convolution while condensing and abstracting the extracted features. A fully connected layer flattens feature maps and establishes full connectivity for tasks such as classification or regression. CNNs consist of convolution, pooling and fully connected layers.

When categorizing CNNs based on object recognition tasks, they are classified into two-stage and single-stage detectors. Two-stage detectors initiate the process by employing a region proposal network (RPN) to generate object proposals. This network produces a region recommendation to determine the potential object region on the image. At this stage, no prediction is made about any object; only possible region suggestions are produced. Then, the feature vectors associated with the recommendation region and the image region are classified using a classifier (usually a CNN). In this phase, object classification and location adjustment are performed on each region proposal. Single-stage detectors perform object classification and location detection directly on the image. There is no suggestion generation step, and the detector directly predicts the object for each pixel. It provides a faster operation compared to two-stage detectors.

Two-stage detectors can provide higher sensitivity and accuracy than single-stage detectors but require more computational cost and processing time.

Region-based CNN (R-CNN): It is the first two-stage detector introduced [80]. The algorithm aims to reduce the computational load and increase the detection speed. Instead of covering all regions of an image, 2000 regions of the image are primarily created with a selective search algorithm. Feature extraction is performed with CNN from the selected regions. Then, extracted features are classified with a Support Vector Machine (SVM) algorithm. RCNN has been used to study the effect of combining thermal and visible images on pedestrian detection during the day and at night [81]. The study's RCNN architecture has been tested with both early and late fusion. It has been observed that image detection takes approximately 47 seconds. This result is not optimal for real-time applications. Classification of 2000 priority regions increases the training time of the R-CNN model.

Spatial Pyramid Pooling (SPP)-Net: This novel approach is introduced to address the limitations observed in the R-CNN algorithm [82]. Employing multiple pooling layers at various scales, this innovative method facilitates the manipulation of input images through cropping and resizing. Notably, it removes the necessity to conform to a specific aspect ratio. Irrespective of input size fluctuations, SPP-Net can produce a constant-length representation. Unlike R-CNN, which individually processes all 2000 regions, SPP-Net operates on the entire image simultaneously, resulting in a marked enhancement in algorithmic processing speed.

Fast-R-CNN is developed to increase the training and testing speed of R-CNN [83]. In the new version, input images are processed, and a convolutional feature map is produced instead of region suggestions being processed by the CNN network. Fast-R-CNN is nine times faster in training and 213 times faster in inference compared to R-CNN.

Faster-R-CNN: While Fast-R-CNN offers notable enhancements in processing speed, it relies on a selective search algorithm to detect regions and their corresponding bounding boxes. This results in a considerable processing delay. To overcome this drawback, a region proposal network (RPN) to estimate bounding boxes has been proposed [84]. This network introduces a model named Faster R-CNN, which integrates with R-CNN to share convolution features. With a test time of just 0.2 seconds, Faster R-CNN establishes itself as a promising choice for real-time applications. Faster R-CNN has successfully performed various tasks, such as pedestrian detection using thermal and colour images in day and night detection scenarios [85]. Single-stage detectors are faster than their two-stage counterparts. This method is more available for real-time development but has lower levels of precision and accuracy.

You Only Look Once (YOLO): Single-stage detectors employ a singular regression step instead of the multi-stage classification process utilized in other methods. YOLO is one of the most popular algorithms of single-stage detectors [86]. In YOLO, the input image is partitioned into a predetermined number of grids. After this, a single neural network estimates bounding boxes and determines their associated class probabilities. These operations are performed in a single step. Although YOLO exhibits greater speed than two-stage detectors, it often leads to higher localization errors and lower detection accuracy when dealing with small objects. These limitations have been improved in YOLOv2, YOLO9000 [87], and YOLOv3 [88]. In [89], YOLO has been used for 3D Lidar and RGB fusion. In another study, YOLOv3 has been used for vision and lidar point cloud fusion [90].

Single-Shot Multibox Detector (SSD): Observations indicate that the YOLO algorithm predominantly detects large objects. According to [91], YOLO's accuracy decreases when dealing with small and variable-scale objects, and according to [92], it imposes spatial constraints on bounding boxes, limiting the classification to a single object class. SSD has been proposed to remove these limitations in YOLO [93]. SSD is engineered to accommodate bounding boxes with diverse sizes and aspect ratios, empowering the algorithm to identify objects of varying dimensions within a single image. SSD has been used for general object detection in autonomous driving applications [94]. In another study, SSD has been used as a primary detector, and various fusion techniques have been compared to multiple CNN architectures [95].

Deconvolutional Single-Shot Detector (DSSD): Since small objects produce a limited number of pixels and information, detecting these objects becomes a burden. Improving accuracy is often prioritized over detection speed [96]. Instead of employing the original Visual Geometry Group (VGG) classifier, DSDD utilizes ResNet101. Consequently, it enhances the SSD algorithm by incorporating deconvolutional layers and enriching the contextual information. DSDD improves the detection of small objects by providing better-resolution feature maps. For pedestrian detection, colour and thermal images are combined using mid-fusion with the DSSD network [97,98].

3.6 Interval analysis methods

Interval analysis methods involve representing sensor measurements as intervals rather than exact values. This approach considers uncertainty and can provide robust fusion results, especially in the presence of sensor noise and errors.

Fault Tolerant Interval Function: This method primarily provides fault tolerance in sensor fusion. It is designed to detect errors in data from different sensors and respond to them tolerantly. For example, it can tolerate missing or incorrect data points or smooth out mismatches between sensor data. Measurement errors, sensor malfunctions, and environmental effects can cause errors in sensor data. The fault-tolerant interval function allows the system to avoid incorrect results caused by incorrect data and works more reliably. The fault tolerant interval function is shown in Equation (21).

$$y(t) = x(t) + e(t) \tag{21}$$

y(t) represents the corrected output, x(t) represents the input, and e(t) represents errors or uncertainties in the data from the sensors. e(t) can change depending on the characteristics of a particular fault tolerant interval function and the methods used.

3.7 Feature maps

Feature maps are representations of sensor data that highlight important features or patterns. They are used to preprocess sensor data before fusion or as input to fusion algorithms.

Occupancy Grid: In sensor fusion, it can be used to combine data from different sensors and extract or update feature maps. It is represented by a matrix. This matrix divides an environment at specific resolutions and represents an occupied or empty state of each cell ($O_{i,j} = \{0,1\}$). $O_{i,j}$ represents the cell in the i,j index of the matrix. 1 means occupied, 0 means empty. The Occupancy grid is updated with various sensor data. For example, environmental sensors such as a LIDAR sensor, radar sensor, or camera can provide data in different directions and resolutions. This data is used to mark specific cells of the occupancy grid as occupied (1) or empty (0).

Network Scans: Sensor fusion provides more comprehensive environmental sensing by combining data from network scanners with data from other sensors. For example, a network scanner can detect IP addresses, connection states, and other network properties of devices, while other sensors can detect physical objects in the environment. Among the most commonly used network scanning techniques is eScan [99]. Escan gathers data within WSNs and generates maps based on the information collected.

4 Conclusion

Sensor fusion integrates data from multiple sensors to acquire comprehensive, reliable, and accurate information. It enhances system performance by ensuring redundancy in case of sensor failure and by combining data from sensors with different sensitivities to better adapt to environmental conditions. Moreover, sensor fusion techniques streamline processing costs by condensing sensor data into fewer, more meaningful datasets.

Algorithms used in sensor fusion applications vary in implementation and effectiveness depending on specific use cases and system requirements. Algorithms such as Kalman filtering and covariance intersection are favoured for their ability to handle Gaussian noise well, making them suitable for applications such as robotics localisation and navigation systems tracking. These algorithms rely on probabilistic models to effectively integrate sensor data and provide robust predictions even in noisy environments. In contrast, knowledge-based algorithms such as fuzzy logic and abductive reasoning are superior in scenarios where sensor data is inherently uncertain. Fuzzy logic enables flexible decision-making based on fuzzy sets and linguistic variables, making it ideal for systems that require human-like reasoning in uncertain situations. Abductive reasoning complements this by extracting the most plausible explanations from incomplete or conflicting sensor inputs. Therefore, it is useful in diagnostic applications such as fault detection systems. Algorithms such as ANNs and CNNs are increasingly preferred because they can learn complex relationships and patterns directly from data. ANNs are preferred for tasks such as object recognition in autonomous vehicles due to their ability to integrate various sensor inputs and generalize from large data sets. On the other hand, CNNs excel at processing image-based sensor data and perform well in applications requiring real-time object detection and classification. Interval analysis algorithms provide robustness against sensor noise and errors by representing sensor measurements as intervals rather than precise values. These algorithms are effective in fault-tolerant systems where maintaining system reliability during sensor failures is critical.

Each algorithm offers unique advantages tailored to specific sensor fusion applications. For example, statistical algorithms are preferred in scenarios that require precise prediction with minimal computational load, while knowledge-based methods are effective for systems that require adaptive decision-making in uncertain environments. Machine learning-based algorithms are particularly valuable in applications that require high accuracy in complex data interpretation. By understanding these strengths, researchers can effectively select the optimal algorithm to optimize system performance and reliability in real-world applications.

In recent years, sensor fusion applications have increasingly embraced the power of artificial intelligence (AI) algorithms. Integrating AI and ML algorithms in sensor fusion applications presents opportunities and challenges. AI/ML's big data analysis capabilities can be used at all levels of sensor fusion. However, combining the condition and impact assessment methods of sensor fusion with AI/ML's big data-based object assessment classification is difficult. For example, although AI/ML has been successful in big data analysis, it still faces challenges for more complex problems such as force structure analysis and intent assessment. AI/ML methods must develop robust models to reduce uncertainty with reliable and consistent performance bounds in this context. The combination of AI/ML and sensor fusion offers significant opportunities in contextual modelling, context awareness methods for situational assessment, and model-based AI/ML methods for threat assessment. Future coordination efforts include the development of contextual models, the ability of AI/ML to transfer learning from one state to another, understanding models derived from principles, and distributed architectures for multi-sensor and multi-algorithm coordination. These efforts aim to increase AI/ML's explainability, ensuring its broader acceptance and effectiveness in sensor fusion applications.

There is a wide range of potential future advances in sensor fusion applications that could further improve functionality and use cases. Integrating AI and ML algorithms allows sensor fusion applications to learn from experience, adapt to new environments and make smarter decisions. Thus, improvements in areas such as navigation and obstacle avoidance are expected to be achieved in the future. In addition, advances in edge computing enable systems to be more efficient in dynamic environments by processing data in real-time, reducing delay, and increasing decision-making speed. Energy efficiency and sustainability advances can contribute to developing systems with longer battery life powered by sustainable power sources, expanding operational capabilities and reducing environmental impact. By evaluating these potential advances and integrating them into the design and development of sensor fusion applications, researchers and engineers can further increase these systems' functionality, performance, and versatility and create new uses across various industries and fields.

Acknowledgment

Hande ÇAVŞİ ZAİM, one of the authors of this article, is supported by the Council of Higher Education of the Republic of Turkey, 100/2000 PhD scholarship in Advanced Robotic Systems and Mechatronics.

Conflict of interest

The author declares that there is no conflict of interest.

Similarity rate (iThenticate): 21%

References

[1] F. E. White, JDL, Data Fusion Lexicon, Technical Panel for C3 15, 1991.

[2] Hall, L. David, J. Llinas. An introduction to multisensor data fusion. Proceedings of the IEEE, 85(1), pp. 6-23, 1997. https://doi.org/10.1109/5.554205

[3] F. Castanedo, A review of data fusion techniques. The Scientific World Journal, 2013(1), 704504, 2013. https://doi.org/10.1155/2013/704504

[4] K. S. Nagla, M. Uddin, D. Singh. Multisensor data fusion and integration for mobile robots: a review. IAES International Journal of Robotics and Automation, 3(2), pp. 131, 2014.

[5] H.F. Durrant-Whyte. Sensor models and multisensor integration. The İnternational Journal of Robotics Research, 7(6), pp. 97-113, 1988. https://doi.org/10.1177/027836498800700608

[6] R.C. Luo, M. G. Kay. A tutorial on multisensor integration and fusion. In Proceedings IECON'90: 16th Annual Conference of IEEE Industrial Electronics Society, pp. 707-722, 1990. https://doi.org/10.1109/IECON.1990.149228

[7] J. Llinas, D. L. Hall, An introduction to multi-sensor data fusion. In: ISCAS'98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, 6, pp. 537-540, May 1998. https://doi.org/10.1109/ISCAS.1998.705329

[8] M. B. Alatise, G. P. Hancke, A review on challenges of autonomous mobile robot and sensor fusion methods. IEEE Access, 8, 39830-39846, 2020. https://doi.org/10.1109/ACCESS.2020.2975643

[9] M. L. Fung, M. Z. Chen, Y. H. Chen, Sensor fusion: A review of methods and applications. In 2017 29th Chinese Control and Decision Conference (CCDC), pp. 3853-3860, IEEE, May 2017. https://doi.org/10.1109/CCDC.2017.7979175

[10] K. Akkaya, I. Guvenc, R. Aygun, N. Pala, A. Kadri, IoT-based occupancy monitoring techniques for energy-efficient smart buildings. In 2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), pp. 58-63, IEEE, March 2015. https://doi.org/10.1109/WCNCW.2015.7122529

[11] N. Ouerhani, N. Pazos, M. Aeberli, M. Muller, IoT-based dynamic street light control for smart cities use cases. In 2016 International Symposium on Networks,

Computers and Communications (ISNCC), pp. 1-5, IEEE, May 2016. https://doi.org/10.1109/ISNCC.2016.7746112

[12] F. Kirsch, R. Miesen, M. Vossiek, Precise local-positioning for autonomous situation awareness in the Internet of Things. In 2014 IEEE MTT-S International Microwave Symposium (IMS2014), pp. 1-4, June 2014. https://doi.org/10.1109/MWSYM.2014.6848674

[13] C. L. Wu, Y. Xie, S. K. Pradhan, L. C. Fu, Y. C. Zeng, Unsupervised context discovery based on hierarchical fusion of heterogeneous features in real smart living environments. In 2016 IEEE International Conference on Automation Science and Engineering (CASE), pp. 1106-1111, IEEE, August 2016. https://doi.org/10.1109/COASE.2016.7743528

[14] S. K. Datta, R. P. F. Da Costa, J. Härri, C. Bonnet, Integrating connected vehicles in Internet of Things ecosystems: Challenges and solutions. In 2016 IEEE 17th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1-6, IEEE, June 2016. https://doi.org/10.1109/WoWMoM.2016.7523574

[15] S. Wildstrom, Better living through big data. Cisco Newsroom. Available: http://newsroom.cisco.com/feature/778800/Better-Living-Through-Big-Data, Accessed 11 March 2024.

[16] W. Ding, X. Jing, Z. Yan, and L. T. Yang, A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion, Information Fusion, vol. 51, pp. 129-144, 2019. https://doi.org/10.1016/j.inffus.2018.12.001

[17] A. Westenberger, M. Muntzinger, M. Gabb, M. Fritzsche, and K. Dietmayer, Time-to-collision estimation in automotive multi-sensor fusion with delayed measurements, in Advanced Microsystems for Automotive Applications 2013: Smart Systems for Safe and Green Vehicles, Springer International Publishing, pp. 13-20, 2013. https://doi.org/10.1007/978-3-319-00476-1_2

[18] X. Wei, Autonomous control system for the quadrotor unmanned aerial vehicle, in 2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), IEEE, pp. 796-799, August 2016. https://doi.org/10.1109/URAI.2016.7733984

[19] H. C. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections, Nature, 293(5828), pp. 133-135, 1981. https://doi.org/10.1038/293133a0

[20] E. M. Upadhyay and N. K. Rana, "Exposure fusion for concealed weapon detection," in 2014 2nd International Conference on Devices, Circuits and Systems (ICDCS), pp. 1-6, IEEE, March 2014. https://doi.org/10.1109/ICDCSyst.2014.6926141

[21] A. M. Sagi-Dolev, Multi-threat detection system, U.S. Patent 8171810, 2012.

[22] J. Favre, B. M. Jolles, O. Siegrist, and K. Aminian, Quaternion-based fusion of gyroscopes and accelerometers to improve 3D angle measurement, Electronics Letters, 42(11), p. 1, 2006. https://doi.org/10.1049/el:20060124

[23] G. Koshmak, A. Loutfi, M. Linden, Challenges and issues in multisensor fusion approach for fall detection. Journal of Sensors, 2016(1), 6931789, 2016. https://doi.org/10.1155/2016/6931789

[24] A. Rihar, M. Mihelj, J. Pašič, J. Kolar, M. Munih, Using sensory data fusion methods for infant body posture assessment. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) pp. 292-297, September 2015. https://doi.org/10.1109/IROS.2015.7353388

[25] S. Knoop, S. Vacek, R. Dillmann, Sensor fusion for 3D human body tracking with an articulated 3D body model. In Proceedings 2006 IEEE International Conference on Robotics and Automation, ICRA, pp. 1686-1691, May 2006. https://doi.org/10.1109/ROBOT.2006.1641949

[26] M. T. Yang, S. Y. Huang, Appearance-based multimodal human tracking and identification for healthcare in the digital home. Sensors, 14(8), 14253-14277, 2014. https://doi.org/10.3390/s140814253

[27] S. Begum, S. Barua, M. U. Ahmed, Physiological sensor signals classification for healthcare using sensor data fusion and case-based reasoning. Sensors, 14(7), 11770-11785, 2014. https://doi.org/10.3390/s140711770

[28] D. J. Yeong, G. Velasco-Hernandez, J. Barry, J. Walsh, Sensor and sensor fusion technology in autonomous vehicles: a review. Sensors, 21(6), 2140, 2021. https://doi.org/10.3390/s21062140

[29] W. Elmenreich, An introduction to sensor fusion. Vienna University of Technology, Austria, 502, pp. 1-28, 2002.

[30] A. Deo, V. Palade, M. N. Huda, Centralised and Decentralised Sensor Fusion-Based Emergency Brake Assist. Sensors, 21(16), 5422, 2021. https://doi.org/10.3390/s21165422

[31] J. Fayyad, M. A. Jaradat, D. Gruyer, H. Najjaran, Deep learning sensor fusion for autonomous vehicle perception and localization: a review. Sensors, 20(15), 4220, 2020. https://doi.org/10.3390/s20154220

[32] M. Almasri, K. Elleithy, Data fusion in WSNs: architecture, taxonomy, evaluation of techniques, and challenges. Int J Sci Eng Res, 6(4), 1620-36, 2015.

[33] K. Huang, B. Shi, X. Li, S. Huang, Y. Li, Multi-modal sensor fusion for auto driving perception: a survey. arXiv preprint arXiv:2202.02703, 2022. https://doi.org/10.48550/arXiv.2202.02703

[34] Y. Cui, R. Chen, W. Chu, L. Chen, D. Tian, Y. Li, D. Cao, Deep learning for image and point cloud fusion in autonomous driving: a review. IEEE Transactions on Intelligent Transportation Systems, 23(2), 722-739, 2021. https://doi.org/10.1109/TITS.2020.3023541

[35] Y. Wang, Q. Mao, H. Zhu, J. Deng, Y. Zhang, J. Ji, Y. Zhang, Multi-modal 3d object detection in autonomous driving: a survey. International Journal of Computer Vision, 131(8), 2122-2152, 2023. https://doi.org/10.1007/s11263-023-01784-z

[36] H. F. Durrant-Whyte, Sensor models and multisensor integration. The İnternational Journal Of Robotics Research, 7(6), 97-113, 1988. https://doi.org/10.1177/027836498800700608
[37] R. Willett, A. Martin, R. Nowak, Backcasting: adaptive sampling for sensor networks. In Proceedings of the 3rd İnternational Symposium On Information Processing İn Sensor Networks, pp. 124-133, April 2004. https://doi.org/10.1145/984622.984641
[38] V. Gupta, R. Pandey, Data fusion and topology control in wireless sensor networks. WSEAS Transactions on Signal Processing, 4(4), 150-172, 2008.
[39] K. Qian, S. Zhu, X. Zhang, L. E. Li, Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 444-453, 2021.
[40] X. Xu, L. Zhang, J. Yang, C. Cao, W. Wang, Y. Ran, M. Luo, A review of multi-sensor fusion slam systems based on 3D LIDAR. Remote Sensing, 14(12), 2835, 2022. https://doi.org/10.3390/rs14122835
[41] J. Alfred Daniel, C. Chandru Vignesh, B. A. Muthu, R. Senthil Kumar, C. B. Sivaparthipan, C. E. M. Marin, Fully convolutional neural networks for LIDAR–camera fusion for pedestrian detection in autonomous vehicle. Multimedia Tools and Applications, 82(16), 25107-25130, 2023. https://doi.org/10.1007/s11042-023-14417-x
[42] E. F. Nakamura, A. A. Loureiro, A. C. Frery, Information fusion for wireless sensor networks: methods, models, and classifications. ACM Computing Surveys (CSUR), 39(3), 9-es, 2007. https://doi.org/10.1145/1267070.1267073
[43] J. Samatha, G. Madhavi, Securesense: enhancing person verification through multimodal biometrics for robust authentication. Scalable Computing: Practice and Experience, *25*(2), 1040-1054, 2024. https://doi.org/10.12694/scpe.v25i2.2524
[44] A. Rehman, T. Saba, M. Kashif, S. M. Fati, S. A. Bahaj, H. Chaudhry, A revisit of internet of things technologies for monitoring and control strategies in smart agriculture. Agronomy, 12(1), 127, 2022. https://doi.org/10.3390/agronomy12010127
[45] B. V. Dasarathy, Sensor fusion potential exploitation-innovative architectures and illustrative applications. Proceedings of the IEEE, 85(1), 24-38, 1997. https://doi.org/10.1109/5.554206
[46] E. Alpaydin, Classifying multimodal data. In The Handbook of Multimodal-Multisensor Interfaces: Signal Processing, Architectures, and Detection of Emotion and Cognition, 2, pp. 49-69, 2018. https://doi.org/10.1145/3107990.3107994
[47] P. A. Neves, J. J. Rodrigues, K. Lin, Data fusion on wireless sensor and actuator networks powered by the ZenSens system. IET communications, 5(12), 1661-1668, 2011. https://doi.org/10.1049/iet-com.2010.0644
[48] C. Zhao, Y. Wang, A new classification method on information fusion of wireless sensor networks. In 2008 International Conference on Embedded Software and Systems Symposia, pp. 231-236, IEEE, July 2008. https://doi.org/10.1109/ICESS.Symposia.2008.73
[49] K. Maraiya, K. Kant, N. Gupta, Study of data fusion in wireless sensor network. In International Conference on Advanced Computing and Communication Technologies, pp. 535-539, 2011.
[50] M. E. Liggins, C. Y. Chong, I. Kadar, M. G. Alford, V. Vannicola, S. Thomopoulos, Distributed fusion architectures and algorithms for target tracking. Proceedings of the IEEE, 85(1), 95-107, 1997. https://doi.org/10.1109/JPROC.1997.554211
[51] E. Waltz, Multisensor Data Fusion, Boston: Artech House, 685, 1990.
[52] E. Azimirad, J. Haddadnia, The comprehensive review on JDL model in data fusion networks: techniques and methods. International Journal of Computer Science and Information Security, 13(1), 53, 2015.
[53] D. E. Moshou, X. E, Pantazi, Data fusion and its applications in agriculture. In Information and Communication Technologies for Agriculture—Theme II: Data, pp. 17-40, Springer International Publishing, 2002. https://doi.org/10.1007/978-3-030-84148-5_2
[54] I. Ermolov, A novel data fusion architecture for uncrewed vehicles. In 21TH International Symposium On Measurement And Control In Robotics ISMCR, pp. 26-28, 2018. https://doi.org/10.21014/acta_imeko.v8i4.684
[55] M. Markin, C. Harris, M. Bernhardt, J. Austin, M. Bedworth, P. Greenway, D. Lowe, Technology foresight on data fusion and data processing. Publication of The Royal Aeronautical Society, 1997.
[56] A. N. Afif, F. Noviyanto, S. Sunardi, S. A. Akbar, E. Aribowo, Integrated application for automatic schedule-based distribution and monitoring of irrigation by applying the waterfall model process. Bulletin of Electrical Engineering and Informatics, 9(1), 420-426, 2020. https://doi.org/10.11591/eei.v9i1.1368
[57] C. Wang, T. Ji, F. Mao, Z. Wang, Z. Li, Prognostics and health management system for electric vehicles with a hierarchy fusion framework: concepts, architectures, and methods. Advances in Civil Engineering, 2021(1), 6685900, 2021. https://doi.org/10.1155/2021/6685900
[58] M. Bevilacqua, A. Tsourdos, A. Starr, I. Durazo-Cardenas, Data fusion strategy for precise vehicle location for intelligent self-aware maintenance systems. In 2015 6th International Conference on Intelligent Systems, Modelling and Simulation, pp. 76-81, IEEE, 2015. https://doi.org/10.1109/ISMS.2015.37
[59] J. R. Boyd, A discourse on winning and losing. Maxwell Air Force Base, AL: Air University Press, 400, 2018.
[60] M. Bedworth, J. O'Brien, The Omnibus model: a new model of data fusion?. IEEE Aerospace and Electronic Systems Magazine, 15(4), 30-36, 2000. https://doi.org/10.1109/62.839632
[61] K. Maraiya, K. Kant, N. Gupta, Study of data fusion in wireless sensor network. In International Conference

on Advanced Computing and Communication Technologies, pp. 535-539, 2011.
[62] R. Alami, R. Chatila, S. Fleury, M. Ghallab, F. Ingrand, An architecture for autonomy. The International Journal of Robotics Research, 17(4), 315-337, 1998. https://doi.org/10.1177/027836499801700402
[63] Y. Tao, H. Hu, H. Zhou, Integration of vision and inertial sensors for 3D arm motion tracking in home-based rehabilitation. The International Journal of Robotics Research, 26(6), 607-624, 2007. https://doi.org/10.1177/0278364907079278
[64] M. A. Lee, B. Yi, R. Martín-Martín, S. Savarese, J. Bohg, Multimodal sensor fusion with differentiable filters. In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 10444-10451, IEEE, October 2020. https://doi.org/10.1109/IROS45743.2020.9341579
[65] Z. Huang, C. Lv, Y. Xing, J. Wu, Multi-modal sensor fusion-based deep neural network for end-to-end autonomous driving with scene understanding. IEEE Sensors Journal, 21(10), 11781-11790, 2020. https://doi.org/10.1109/JSEN.2020.3003121
[66] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, A.Geiger, Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022. https://doi.org/10.1109/TPAMI.2022.3200245
[67] Y. Xiao, F. Codevilla, A. Gurram, O. Urfalioglu, A. M. López, Multimodal end-to-end autonomous driving. IEEE Transactions on Intelligent Transportation Systems, 23(1), 537-547, 2020. https://doi.org/10.1109/TITS.2020.3013234
[68] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. L. Rus, S. Han, Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In 2023 IEEE international conference on robotics and automation (ICRA), pp. 2774-2781, IEEE, 2023. https://doi.org/10.1109/ICRA48891.2023.10160968
[69] A. Singh, Transformer-based sensor fusion for autonomous driving: A survey. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3312-3317, 2023.
[70] K. Lin, Y. Li, J. Sun, D. Zhou, Q. Zhang, Multi-sensor fusion for body sensor network in medical human–robot interaction scenario. Information Fusion, 57, pp. 15-26, 2020. https://doi.org/10.1016/j.inffus.2019.11.001
[71] M. Z. Uddin, M. M. Hassan, A. Alsanad, C. Savaglio, A body sensor data fusion and deep recurrent neural network-based behavior recognition approach for robust healthcare. Information Fusion, 55, 105-115, 2020. https://doi.org/10.1016/j.inffus.2019.08.004
[72] S. Chung, J. Lim, K. J. Noh, G. Kim, H. Jeong, Sensor data acquisition and multimodal sensor fusion for human activity recognition using deep learning. Sensors, 19(7), 1716, 2019. https://doi.org/10.3390/s19071716
[73] G. Shi, C. Geng, H. Liu, H. Su, Y. Jin, S. Sun, The human body characteristic parameters extraction and disease tendency prediction based on multi-sensing fusion algorithms. In 2016 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp. 126-130, IEEE, June 2016. https://doi.org/10.1109/CYBER.2016.7574808
[74] A. Alofi, A. Alghamdi, R. Alahmadi, N. Aljuaid, M. Hemalatha, A review of data fusion techniques. International Journal of Computer Applications, 167(7), pp. 37-41, 2017. https://doi.org/10.5120/ijca2017914318
[75] Q. Li, C. Zhang, Q. Hu, H. Fu, P. Zhu, Confidence-aware fusion using dempster-shafer theory for multispectral pedestrian detection. IEEE Transactions on Multimedia, 2022. https://doi.org/10.1109/TMM.2022.3160589
[76] M. O. Oloyede, G. P. Hancke, Unimodal and multimodal biometric sensing systems: a review. IEEE Access, 4, 7532-7555, 2016. https://doi.org/10.1109/ACCESS.2016.2614720
[77] K. Khoshelham, S. Nedkov, C. Nardinocchi, A comparison of Bayesian and evidence-based fusion methods for automated building detection in aerial data. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 37(PART B7), 1183-1188, 2008.
[78] M. Z. Uddin, M. M. Hassan, A. Alsanad, C. Savaglio, A body sensor data fusion and deep recurrent neural network-based behaviour recognition approach for robust healthcare. Information Fusion, 55, 105-115, 2020. https://doi.org/10.1016/j.inffus.2019.08.004
[79] S. Münzner, P. Schmidt, A. Reiss, M. Hanselmann, R. Stiefelhagen, R. Dürichen, CNN-based sensor fusion techniques for multimodal human activity recognition. In Proceedings of the 2017 ACM International Symposium On Wearable Computers, pp. 158-165, September 2017. https://doi.org/10.1145/3123021.3123046
[80] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition, pp. 580-587, 2014. https://doi.org/10.1109/CVPR.2014.81
[81] J. Wagner, V. Fischer, M. Herman, S. Behnke, Multispectral Pedestrian Detection using Deep Fusion Convolutional Neural Networks. In ESANN, 587, pp. 509-514, April 2016.
[82] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions On Pattern Analysis And Machine İntelligence, 37(9), 1904-1916, 2015. https://doi.org/10.1109/TPAMI.2015.2389824
[83] R. Girshick, Fast r-cnn. In Proceedings of the IEEE İnternational Conference On Computer Vision, pp. 1440-1448, 2015. https://doi.org/10.1109/ICCV.2015.169

[84] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions On Pattern Analysis And Machine İntelligence, 39(6), 1137-1149, 2016. https://doi.org/10.1109/TPAMI.2016.2577031

[85] J. Liu, S. Zhang, S. Wang, D. N. Metaxas, Multispectral deep neural networks for pedestrian detection. arXiv preprint arXiv:1611.02644, 2016. https://doi.org/10.48550/arXiv.1611.02644

[86] J., Redmon, S. Divvala, R. Girshick, A. Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision And Pattern Recognition (pp. 779-788), 2016. https://doi.org/10.1109/CVPR.2016.91

[87] J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision And Pattern Recognition, pp. 7263-7271, 2017. https://doi.org/10.1109/CVPR.2017.690

[88] J. Redmon, A. Farhadi, Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018. https://doi.org/10.48550/arXiv.1804.02767

[89] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, U. J. Nunes, Multimodal vehicle detection: fusing 3D-LIDAR and color camera data. Pattern Recognition Letters, 115, 20-29, 2018. https://doi.org/10.1016/j.patrec.2018.04.011

[90] H. Wang, X. Lou, Y. Cai, Y. Li, L. Chen, Real-time vehicle detection algorithm based on vision and lidar point cloud fusion. Journal of Sensors, 2019. https://doi.org/10.1155/2019/8473980

[91] J. Dou, J. Fang, T. Li, J. Xue, Boosting cnn-based pedestrian detection via 3d lidar fusion in autonomous driving. In Image and Graphics: 9th International Conference, ICIG 2017, Shanghai, China, September 13-15, 2017, Revised Selected Papers, Part II 9, pp. 3-13, Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-71589-6_1

[92] J. Han, Y. Liao, J. Zhang, S. Wang, S. Li, Target fusion detection of LiDAR and camera based on the improved YOLO algorithm. Mathematics, 6(10), 213, 2018. https://doi.org/10.3390/math6100213

[93] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, A. C. Berg, Ssd: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14, Proceedings, Part I 14, pp. 21-37, Springer International Publishing, October 2016. https://doi.org/10.1007/978-3-319-46448-0_2

[94] J. Kim, J. Choi, Y. Kim, J. Koh, C. C. Chung, J. W. Choi, Robust camera lidar sensor fusion via deep gated information fusion network. In 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 1620-1625, IEEE, June 2018. https://doi.org/10.1109/IVS.2018.8500711

[95] Y. L. Hou, Y. Song, X. Hao, Y. Shen, M. Qian, H. Chen, Multispectral pedestrian detection based on deep convolutional neural networks. Infrared Physics & Technology, 94, 69-77, 2018. https://doi.org/10.1016/j.infrared.2018.08.029

[96] Z. Li, L. Yang, F. Zhou, FSSD: feature fusion single shot multibox detector. arXiv preprint arXiv:1712.00960, 2017. https://doi.org/10.48550/arXiv.1712.00960

[97] C. Y. Fu, W. Liu, A. Ranga, A. Tyagi, A. C. Berg, Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659, 2017. https://doi.org/10.48550/arXiv.1701.06659

[98] Y. Lee, T. D. Bui, J. Shin, Pedestrian detection based on deep fusion network using feature correlation. In 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 694-699, IEEE, November 2018. https://doi.org/10.23919/APSIPA.2018.8659688

[99] Y. Zhao, R. Govindan, D. Estrin, Residual energy scans for monitoring wireless sensor networks. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'02), IEEE, Orlando, pp. 356 362, 2002. https://doi.org/10.1109/WCNC.2002.993521