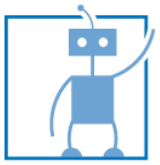


Robotics & Embedded Systems

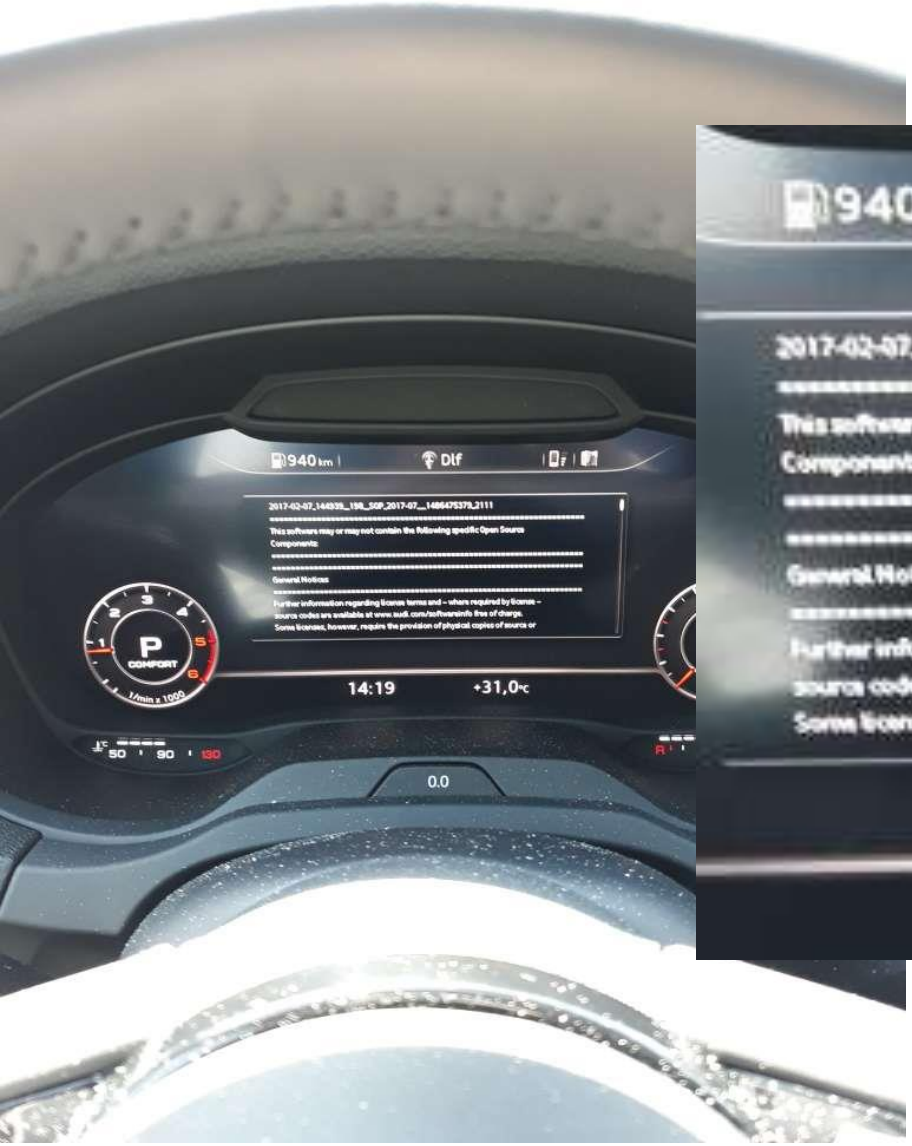
Autonomes Fahren

SS 2019

Multi-Sensor-Data-Fusion

Technische Universität München

Curiosities – Seen in A3 today



Multi-Sensor-Data-Fusion Definition

- Data Fusion Handbook:

  "*Data Fusion is the process for combining data to estimate entity states, where an entity can be any aspect of reality at any degree of abstraction*"

- JDL: *"A multi-level process dealing with the association, correlation, combination of data and information from single and multiple sources to achieve refined position, identify estimates and complete and timely assessments of situations, threats and their significance."*

- Hall & Linas: *"data fusion techniques combine data from multiple sensors and related information from associated databases to achieve improved accuracy and more specific inferences than could be achieved by the use of a single sensor alone."*

- The goal is to transition from conflicting data to reliable information, to obtain a lower detection error probability and a higher reliability by using data from multiple distributed sources.

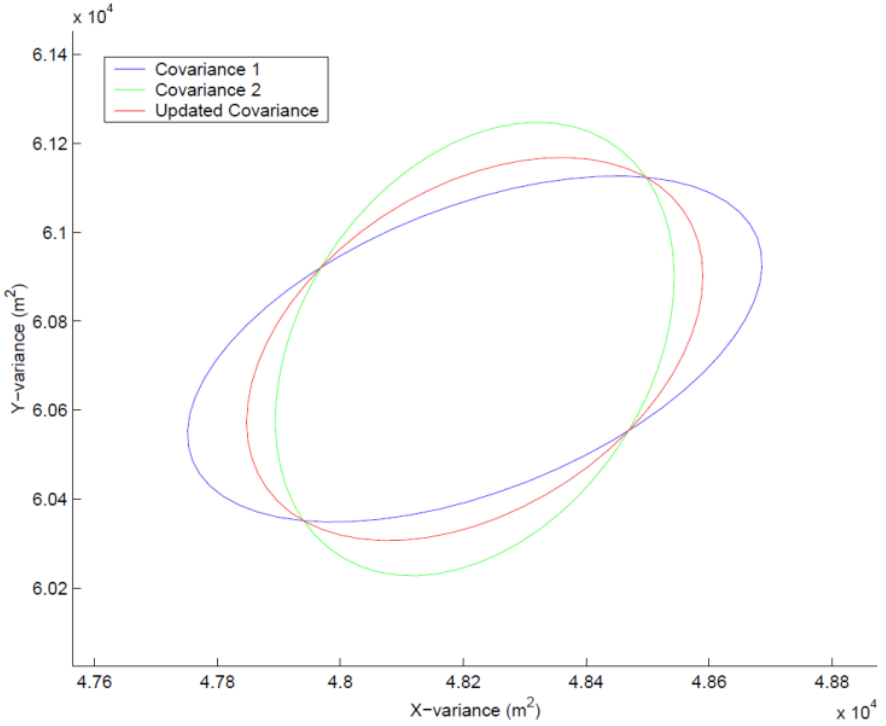
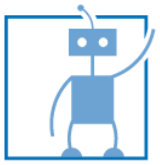


Image source https://www.sto.nato.int/ context enhanced information fusion



“Data Fusion Secrets”

These are some often stated, well known common sense rules for data fusion issues:

- There is no substitute for a good sensor.
- Downstream processing cannot absolve the sins of upstream processing.
- The fused answer may be worse than the best sensor.
- There are no magic algorithms.
- There will never be enough training data.
- It is difficult to quantify the value of data fusion.
- Fusion is not a static process.

verkackte daten können nicht wieder gut gemacht werden

From Data to Information

- Data: Raw signals, numbers, letters symbols, strings, colors…
- Information: Higher semantic value
- Earlier situation: Not enough data and not enough information
- Today: Big data and not enough information

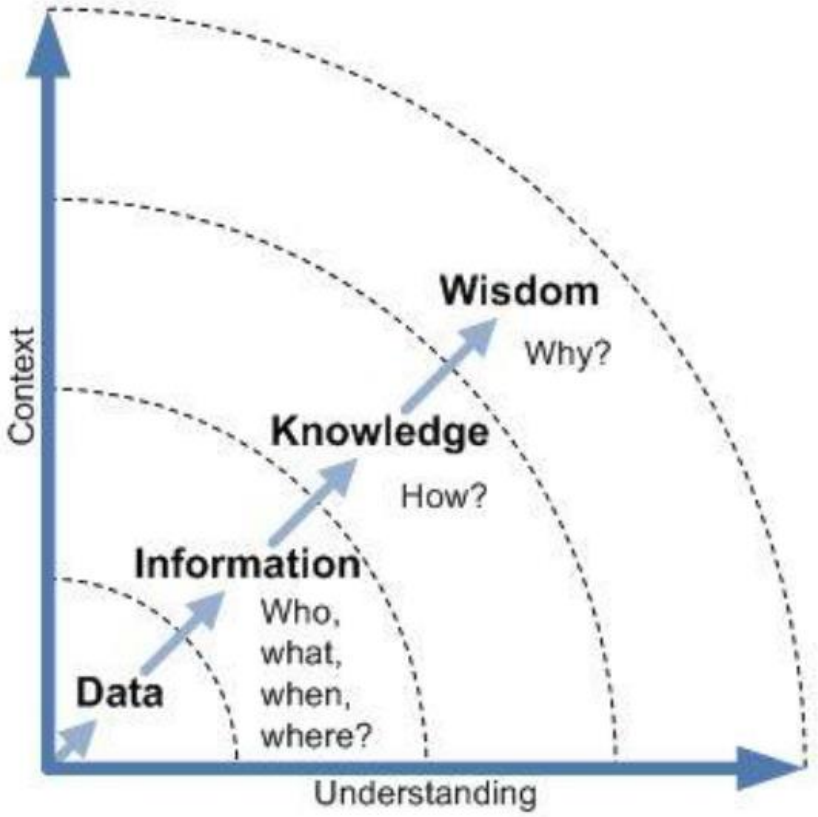


Image source https://www.sto.nato.int/ context enhanced information fusion

From Data to Information

- *Information fusion* and *data fusion* are often employed as synonyms
- Sometimes *data fusion* is used for raw data (obtained directly from the sensors)
- Sometimes *information fusion* is employed to define already processed data.
- *Information fusion* implies a higher semantic level than *data fusion*.
- Other terms associated with data fusion:
  - decision fusion
  - data combination
  - data aggregation,
  - multisensor data fusion
  - sensor fusion

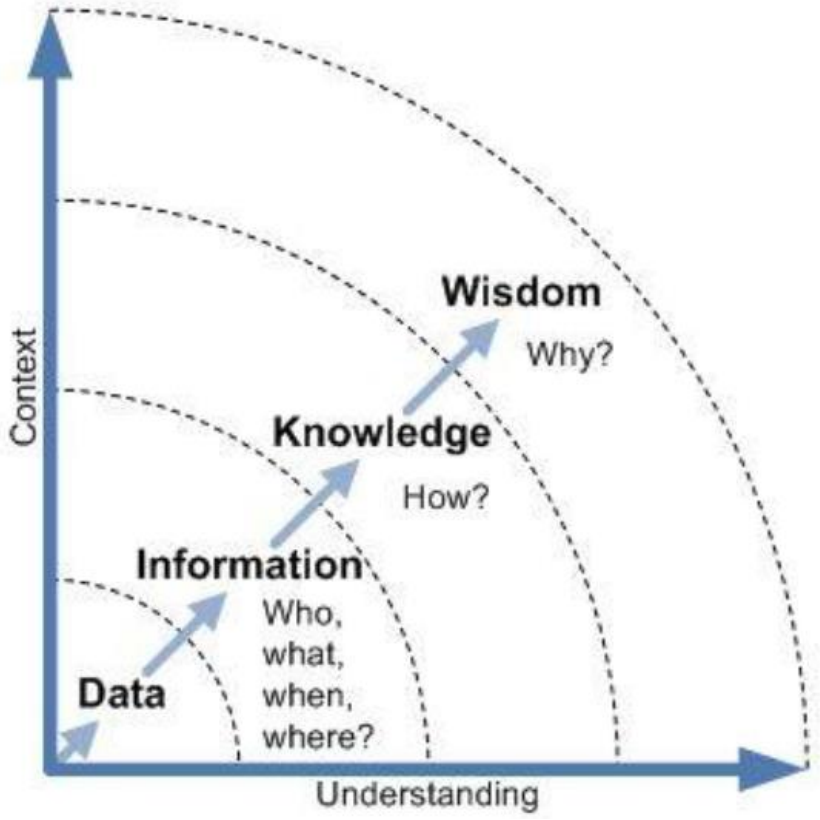


Image source https://www.sto.nato.int/ context enhanced information fusion

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

From Data to Information

- Data fusion techniques have been extensively employed in multisensor environments with the aim of fusing and aggregating data from different sensors.
- These techniques can also be applied to other domains, such as text processing.

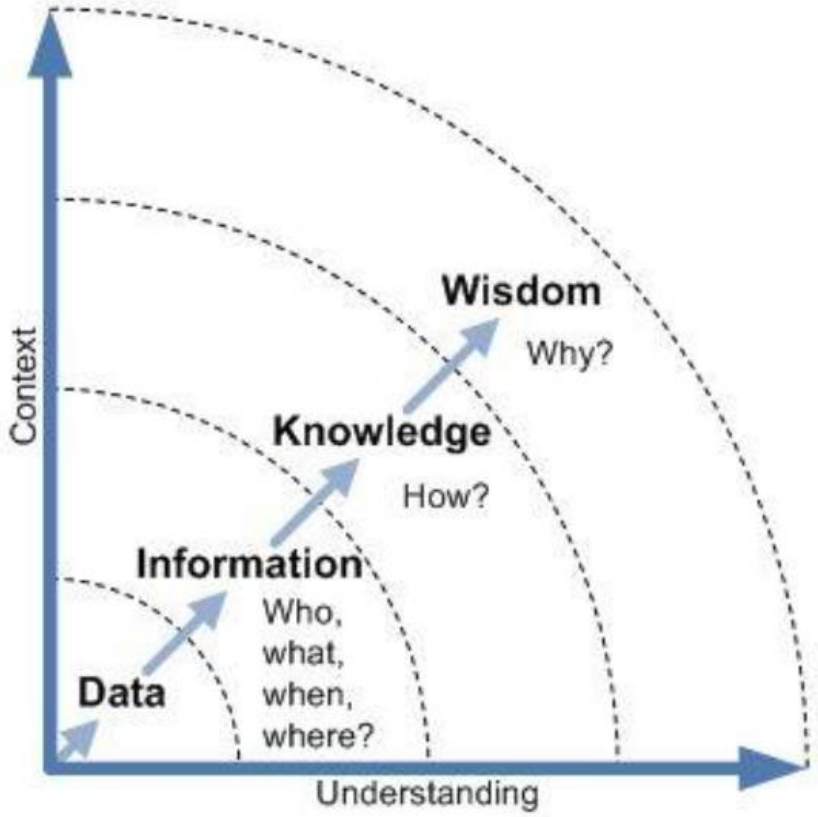
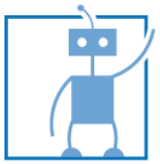


Image source https://www.sto.nato.int/ context enhanced information fusion

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Hierarchical approach to Data Fusion

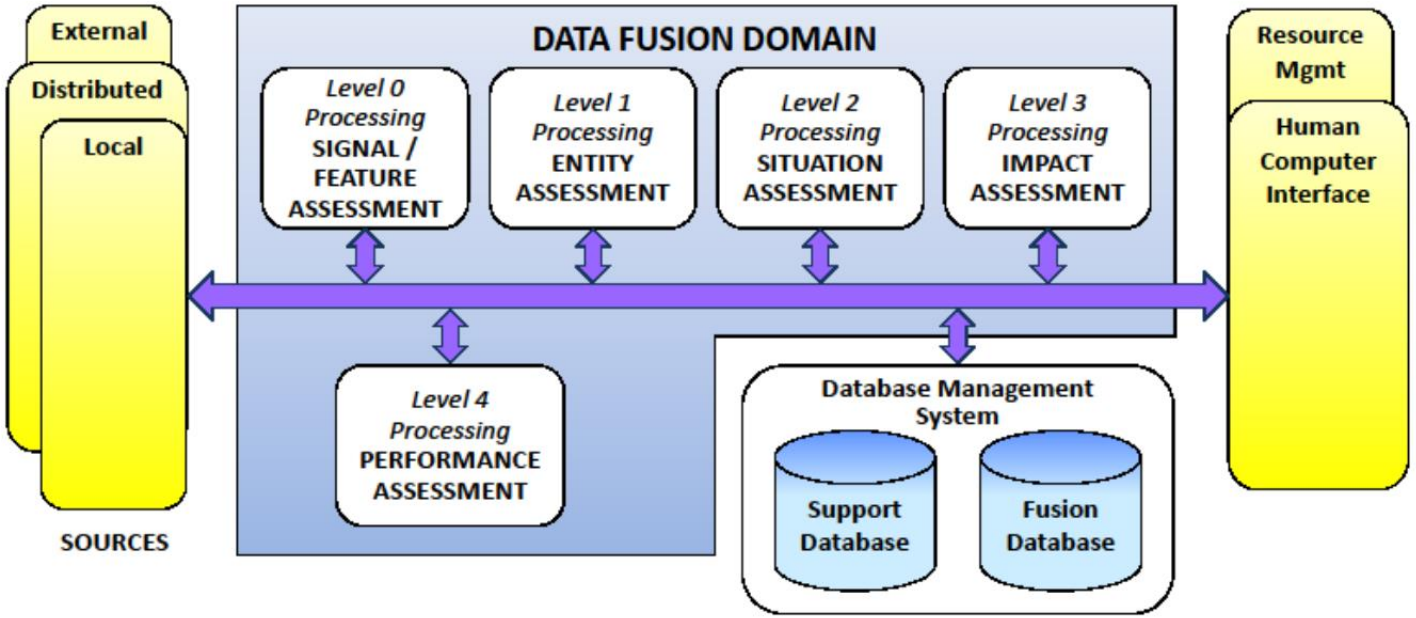
- Level 0—Sub-Object Data Association and Estimation: pixel/signal level data association and characterization
- Level 1—Object Refinement: observation-to-track association, continuous state estimation (e.g. kinematics) and discrete stateestimation (e.g. target type and ID) and prediction
- Level 2—Situation Refinement: object clustering and relational analysis, to include force structure and cross force relations, communications, physical context, etc.
- Level 3—Significance Estimation[Threat Refinement]: threat intent estimation, [event prediction], consequence prediction, susceptibility and vulnerability assessment
- Level 4—Process Refinement: adaptive search and processing (an element of resource management)

Sources: JDL model (1987-91) and the draft revised model (1997), A. Steinberg, adapted from David L. Hall

Hierarchical approach to Data Fusion

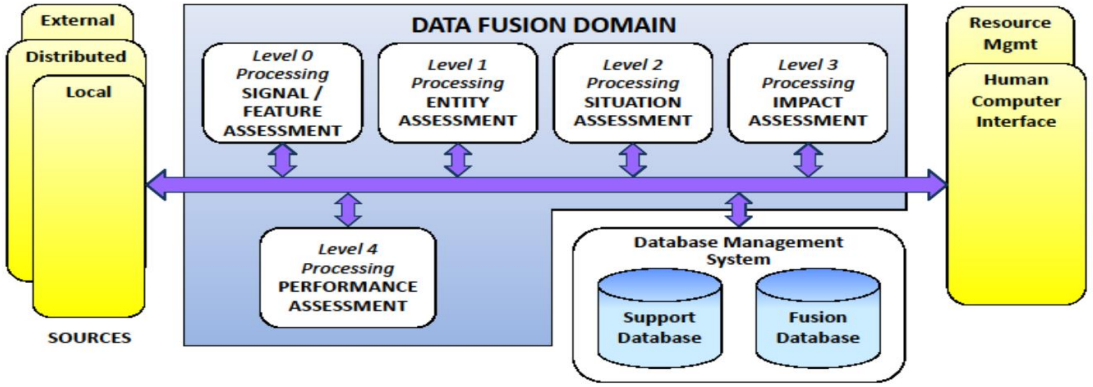
- Level 0—Sub-Object Data Association and Estimation: pixel/signal level data association and characterization
- Level 1—Object Refinement: observation-to-track association, continuous state estimation (e.g. kinematics) and discrete stateestimation (e.g. target type and ID) and prediction
- Level 2—Situation Refinement: object clustering and relational analysis, to include force structure and cross force relations, communications, physical context, etc.
- Level 3—Significance Estimation[Threat Refinement]: threat intent estimation, [event prediction], consequence prediction, susceptibility and vulnerability assessment
- Level 4—Process Refinement: adaptive search and processing (an element of resource management)

JDL model (1987-91) and the draft revised model (1997), A. Steinberg, adapted from David L. Hall, image nato sto.



Hierarchical approach to Data Fusion

- Example of a JDL implementation from NATO STO IST-155.
- A park area monitoring data fusion process is realized from JDL levels 0 to 4.



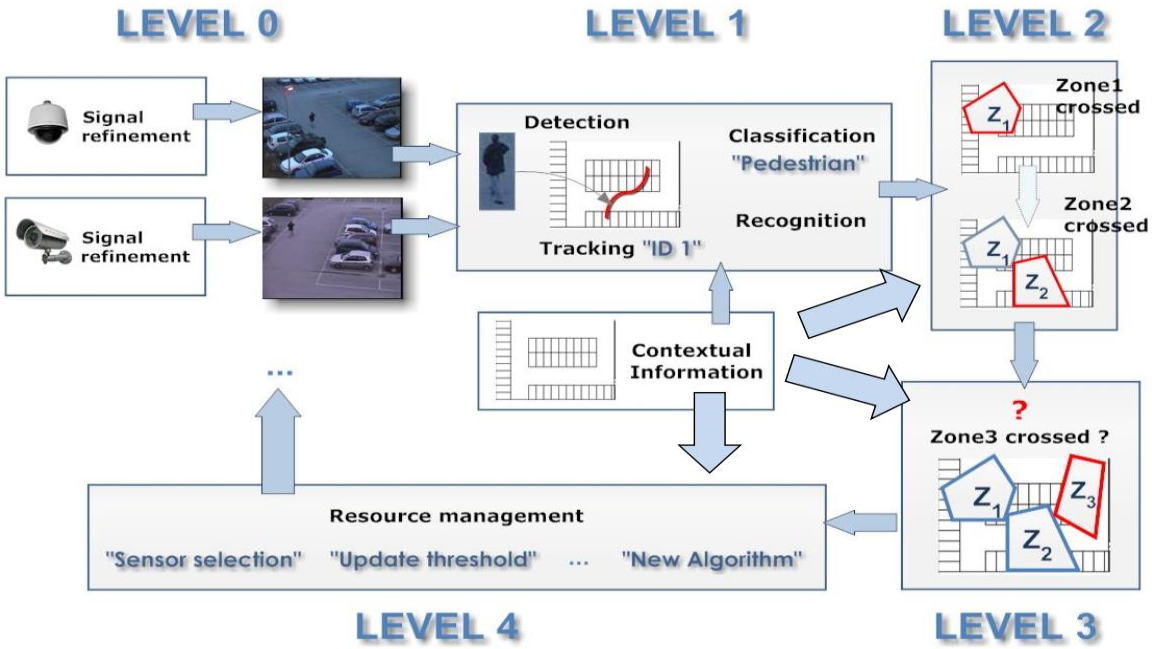
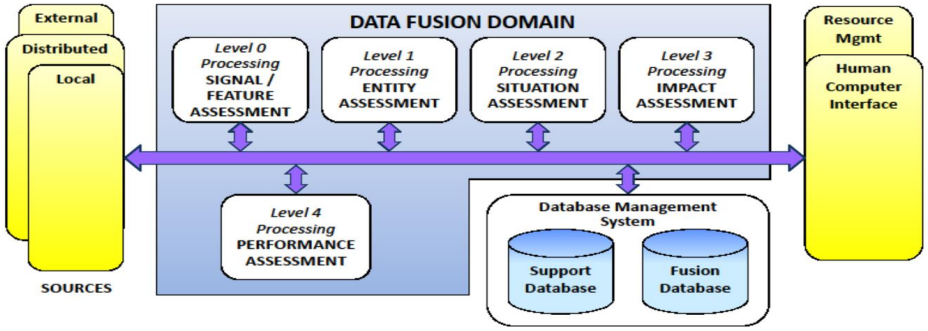


Image source https://www.sto.nato.int/
context enhanced information fusion

Context impact on MSDF

- Context influences the outcome of situations.
- Including context in MSDF approaches, can improve performance.
- Context can be sorted according to JDL levels.
- Further examples include using traffic speed signs to update tracker parameters (JDL level 1), or setting rules for decision support (JDL 2/3) if a lane becomes reduced to truck driving only. Or using events, such as football games to influence parameters.

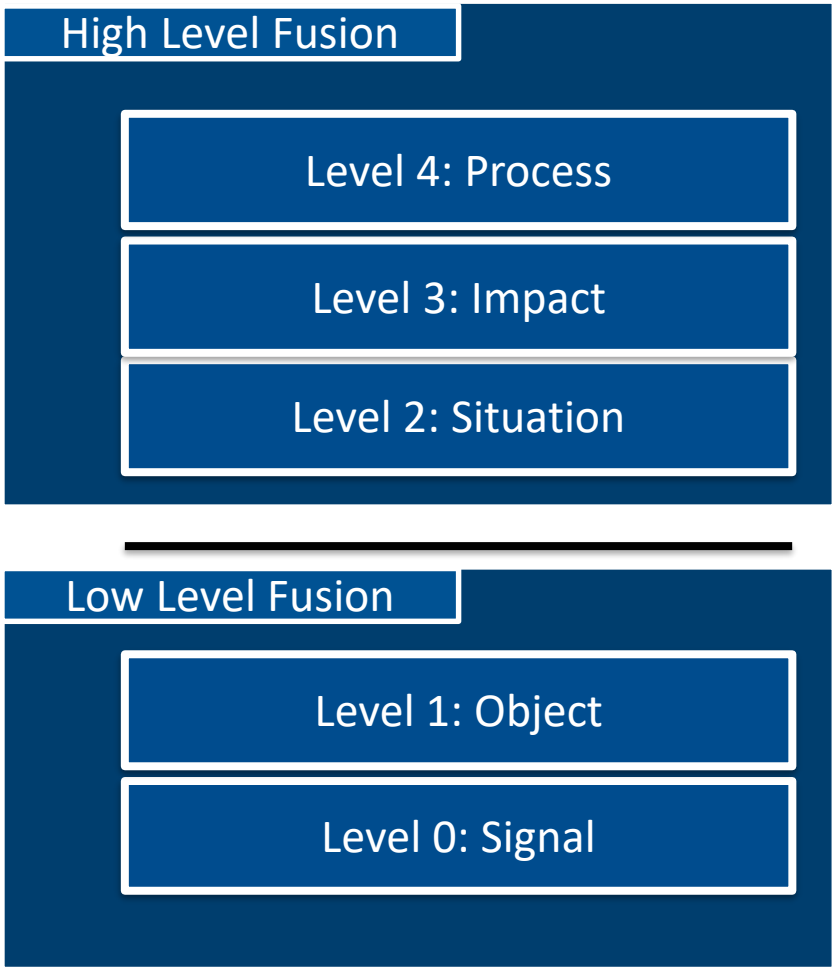


Level	Context	Effect
0	Rainy / Darkness	Reduced camera signal quality
1	Rainy weather	Reduced objects' speed
2	Rainy weather	Traffic more intense
3	Clear / full daylight	Reduced likelihood of car thefts

Image source https://www.sto.nato.int/ context enhanced information fusion

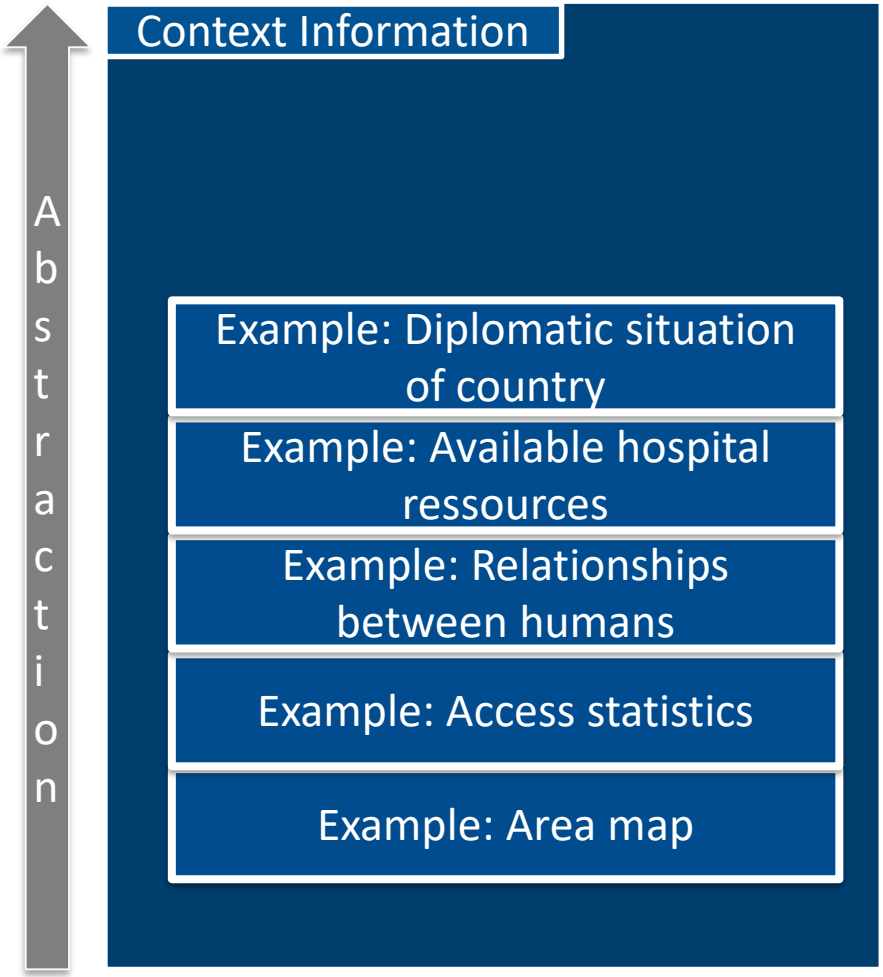
Low Level and High Level Fusion

- Different JDL levels typically work on different dimensionalities. The signal level is relatively low dimensional.
- Starting from level 2, the relations between entities become less and less observable, and worse understood.
- Example: One car performing a lane change, trying to negotiate the maneuver with another car, what's their mathematical relation for the fusion setup? A single object moving forward in a free space is more straight forward.



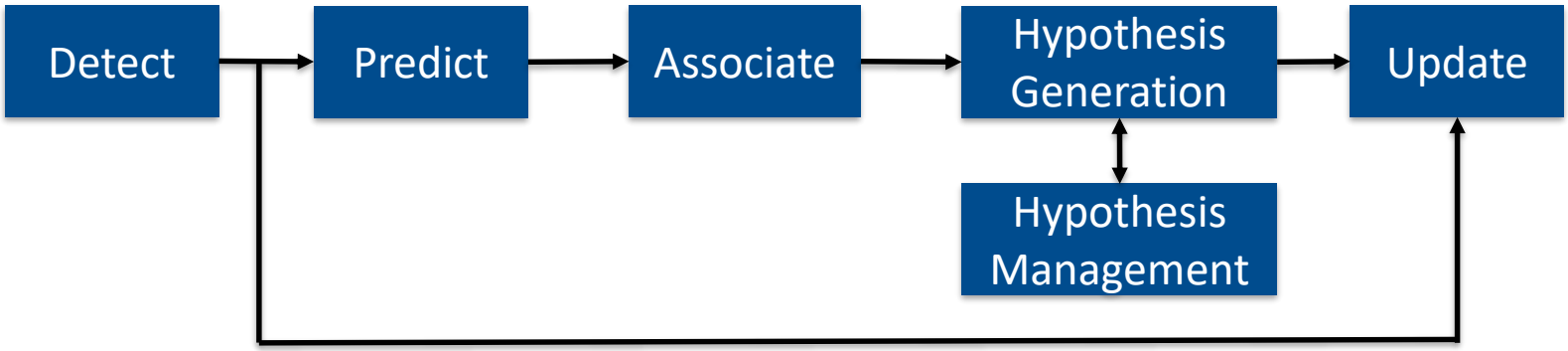
Low Level and High Level Context

- Different JDL levels typically work on different dimensionalities. The signal level is relatively low dimensional.
- Starting from level 2, the relations between entities become less and less observable, and worse understood.
- Example: One car performing a lane change, trying to negotiate the maneuver with another car, what's their mathematical relation for the fusion setup? A single object moving forward in a free space is more straight forward.



2 Level Architecture based on Bowman Model

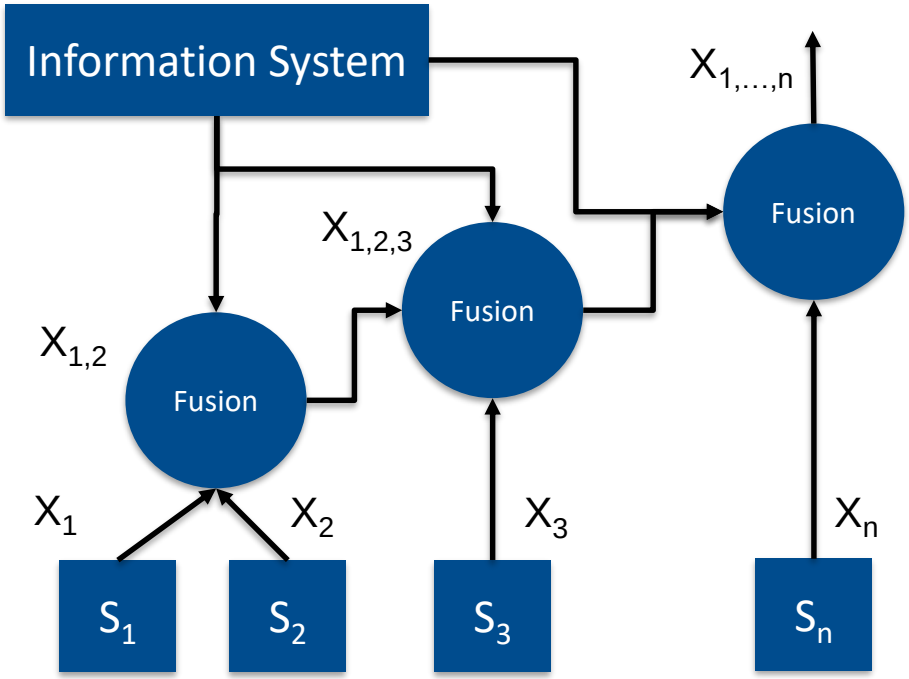
- Bowman architecture (1980).
- Bowman argued that JDL model is useful, but has not helped in developing architecture for a real system.
- Developed the concept of a hierarchical tree of data fusion to divide fusion problems into nodes.
- Conceptually, each node involved the functions as a data link, estimation and correlation.

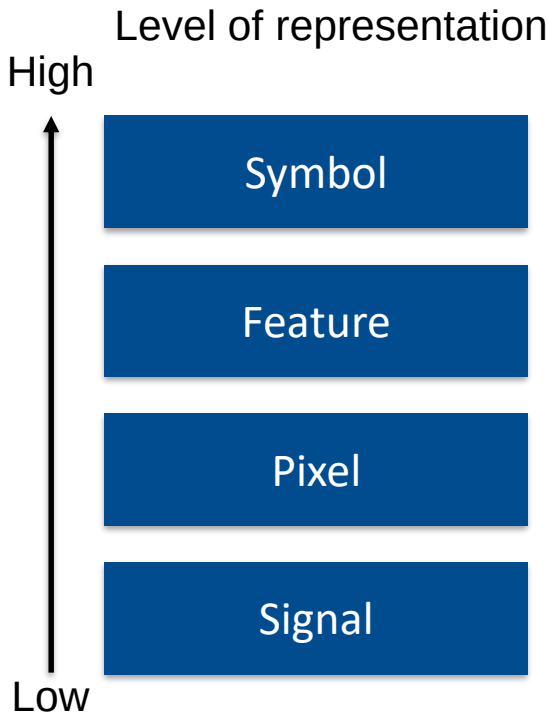


A COMPREHENSIVE REVIEW OF THE MULTI-SENSOR DATA FUSION ARCHITECTURES, EHSAN AZIMIRAD et al., Journal of Theoretical and Applied Information Technology, 2015

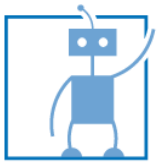
Luo and Kay Architecture

- Luo and Kay suggested a structure for general data fusion based on multi-sensor integration (1988).
- Data from multiple sources is combined inside fusion centres in a hierarchical manner.
- Multi-sensor integration and multi-sensor fusion in four levels: signal, pixel, feature and symbol levels
- Sensor data is transmitted to the fusion centers in hierarchical and sequential manner. Data representation levels are increased the raw data to the decision level.



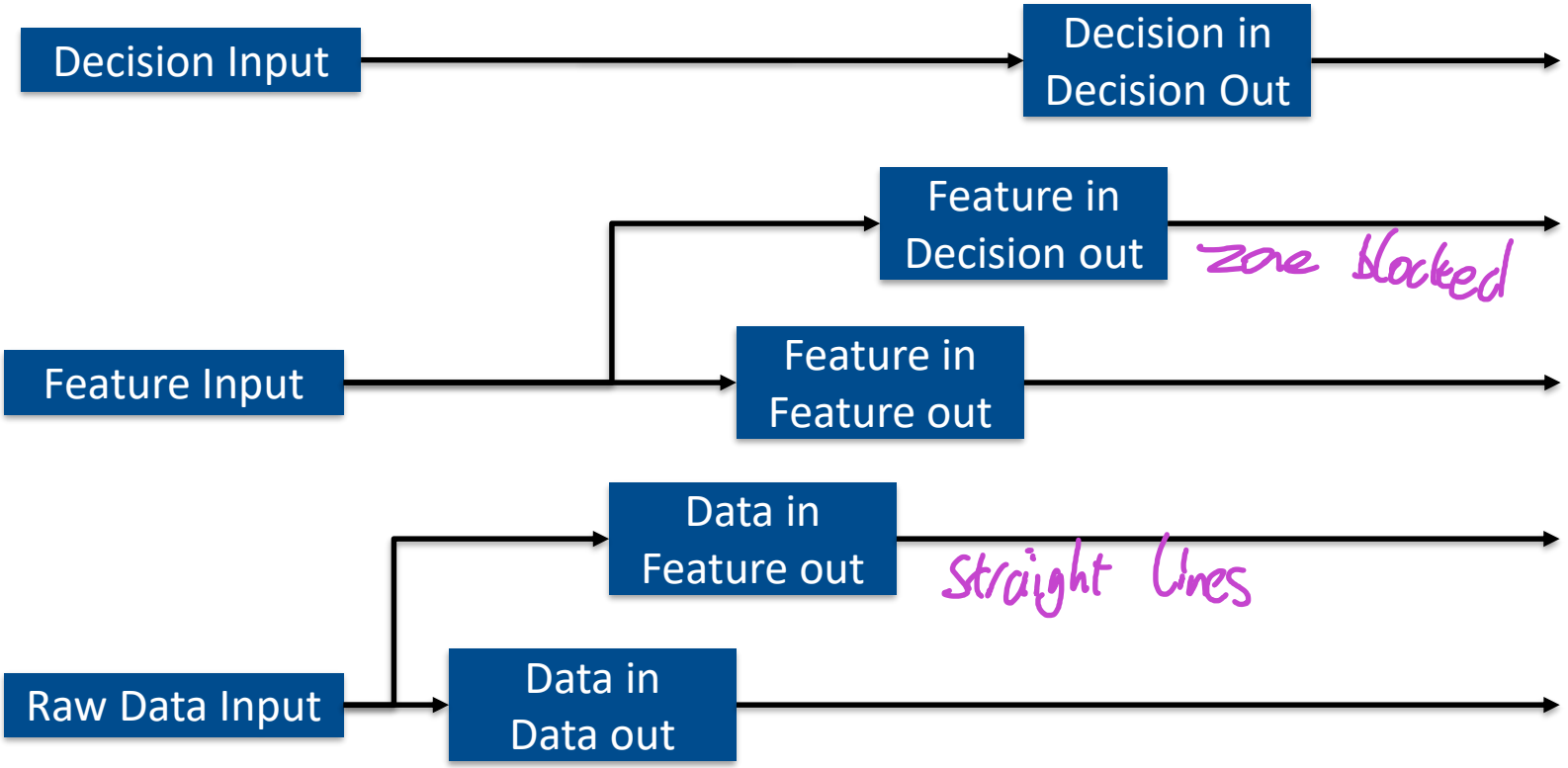


A COMPREHENSIVE REVIEW OF THE MULTI-SENSOR DATA FUSION ARCHITECTURES, EHSAN AZIMIRAD et al., Journal of Theoretical and Applied Information Technology, 2015



Durant-Whyte

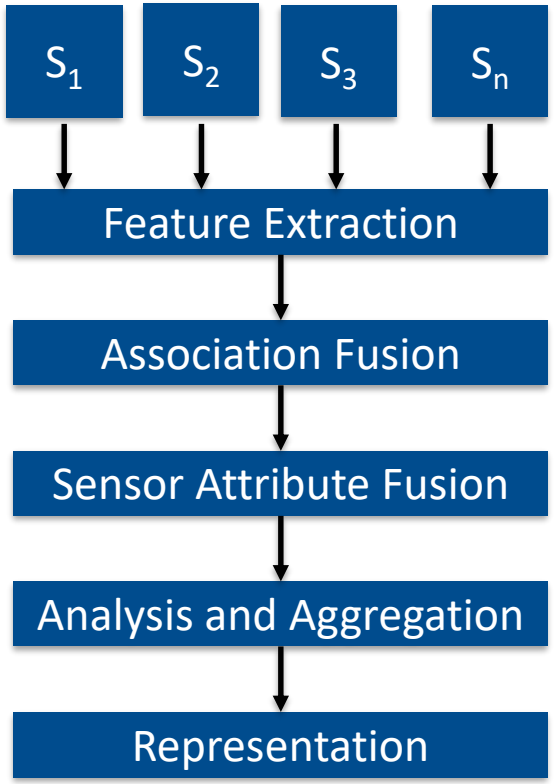
- Durrant-Whyte architecture for robotic systems (1988).
- Main characteristic of this model is the use of a common presentation format.
- Data obtained from the sensors is converted to this format by a combination of high level models.



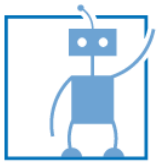
A COMPREHENSIVE REVIEW OF THE MULTI-SENSOR DATA FUSION ARCHITECTURES, EHSAN AZIMIRAD et al., Journal of Theoretical and Applied Information Technology, 2015

Pau Architecture

- Fusion architecture introduced by Pau (1988)
- Hierarchical architecture, providing a model for combining data based on behavioral knowledge.
- The hierarchical approach is composed of three levels:
  - Lowest level: for each sensor there is a vector space with coordinate dimensions and measuring parameters.
  - The next level extracts appropriate features of the vectors, connecting and labeling them.
  - The third level relates feature vectors to events and defines the model of the environment.
- A feature vector is extracted from the raw data, then this vector is aligned and associated with defined attributes. The combination is performed at the attribute sensor level and data analysis is performed subsequently. Finally, a series of behavioral rules can be extracted.

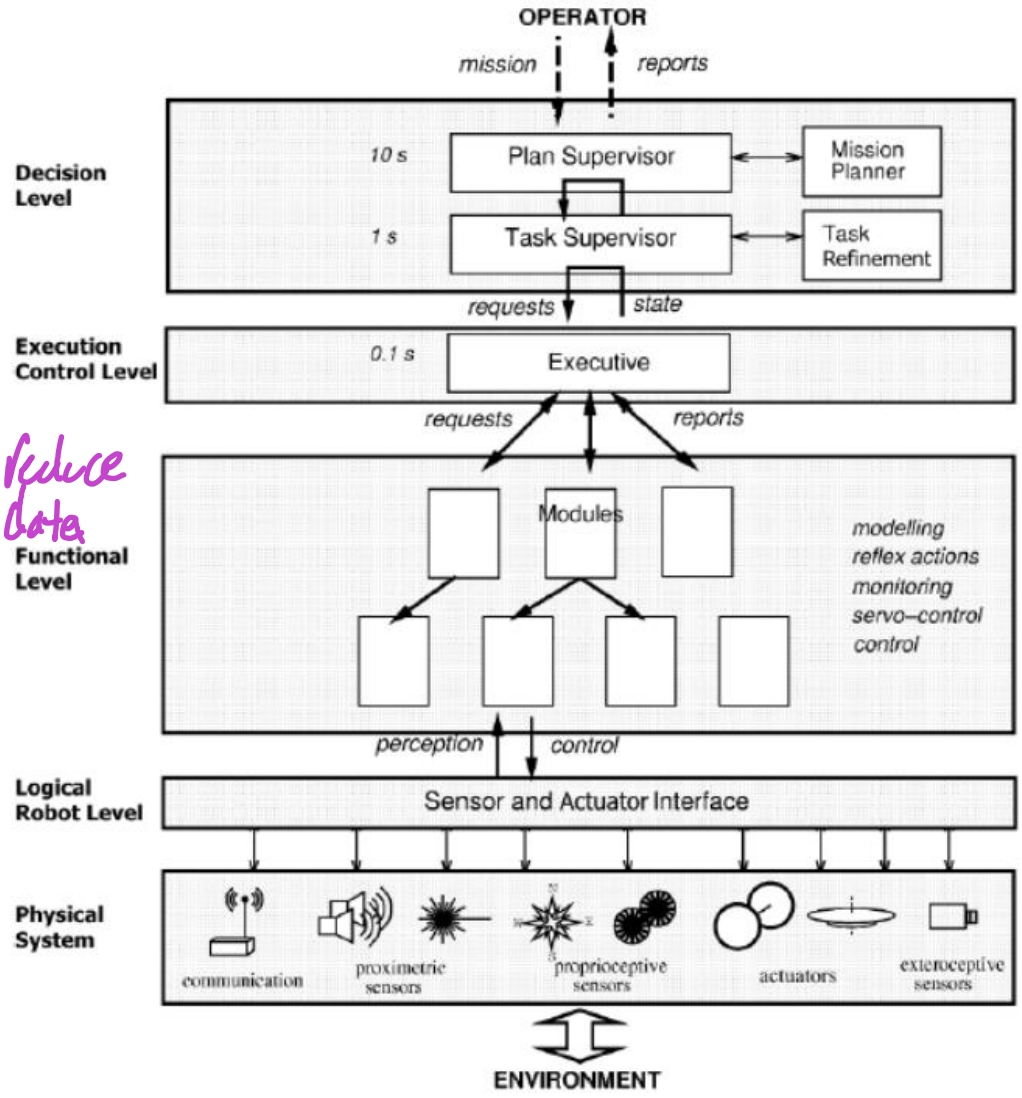


A COMPREHENSIVE REVIEW OF THE MULTI-SENSOR DATA FUSION ARCHITECTURES, EHSAN AZIMIRAD et al., Journal of Theoretical and Applied Information Technology, 2015

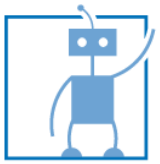


Laas Architecture

- LAAS (Laboratory Analysis Architecture Systems) architecture introduced in 1998.
- Integrated architecture for a real-time mobile robot. LAAS
- Designed for sensor fusion and mid-level classification of functional modules.
- **1) Logical Robot Level:** relates hardware, physical sensors, actuators and functional surface.
- **2) Functional Level:** includes all the features of the conceptual and practical building robot. Arithmetic operations such as image processing, obstacle avoidance and control loop to control the communication module can be classified separately.
- **3) Level Control:** provides the functions related to control and coordinate the work requirements.
- **4) Decision Level:** includes tasks and motion and monitors its implementation. Timing requirements exist at the decision level and the various functional levels.



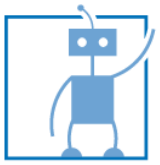
A COMPREHENSIVE REVIEW OF THE MULTI-SENSOR DATA FUSION ARCHITECTURES, EHSAN AZIMIRAD et al., Journal of Theoretical and Applied Information Technology, 2015



Classifying Data Fusion Methods

- Data fusion methods can be classified according to multiple criteria:
  - relations between the input data sources, as proposed by Durrant-Whyte.
    - (a) complementary, (b) redundant, or (3) cooperative data;
  - input/output data types and their nature, as proposed by Dasarathy
  - abstraction level of the employed data:
    - (a) raw measurement, (b) signals, and (c) characteristics or decisions;
  - (4) based on the different data fusion levels defined by the JDL;
  - (5) Depending on the architecture type:
    - (a) centralized, (b) decentralized, or (c) distributed

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

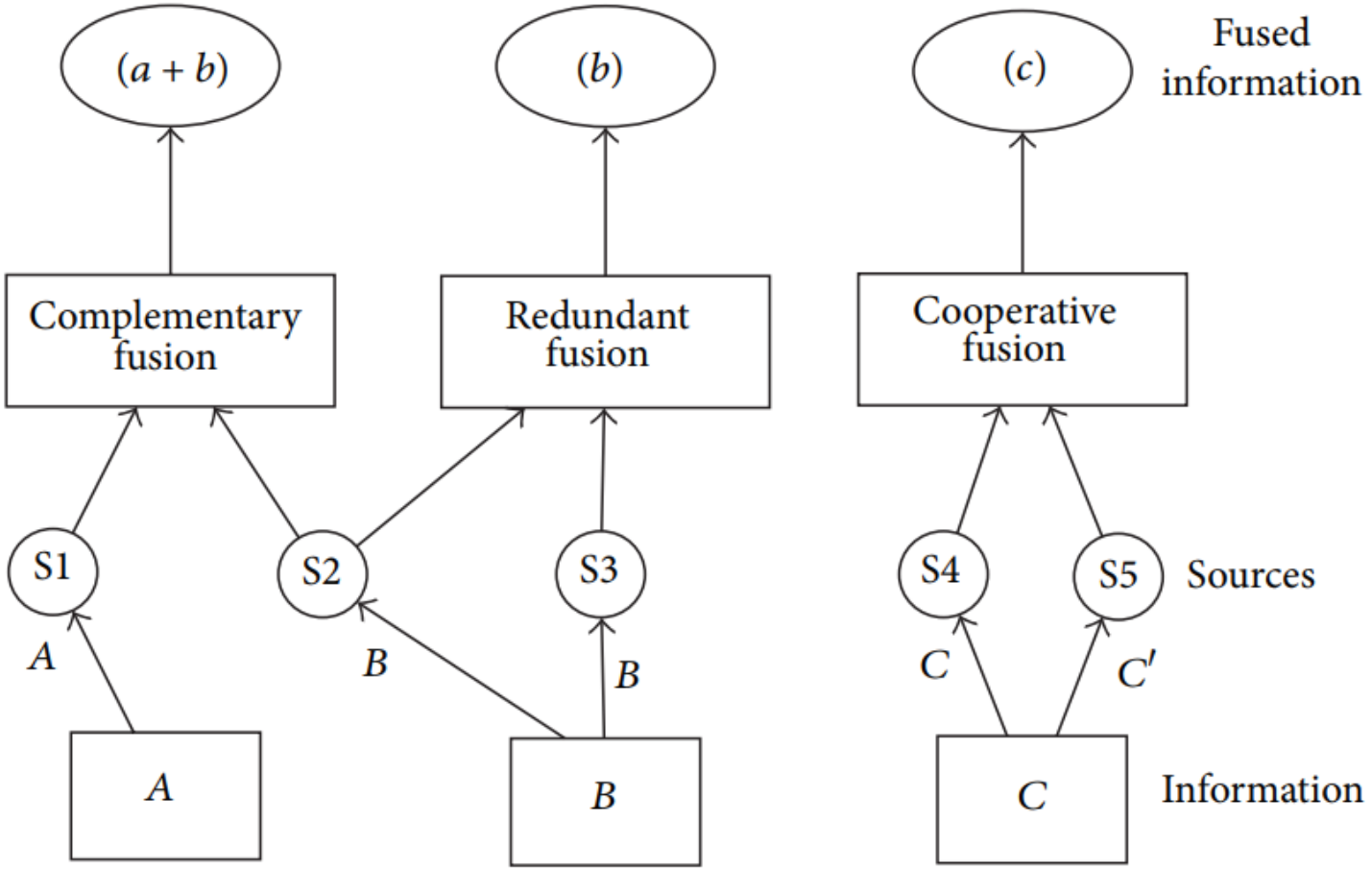


Classification Based on the Relations between the Data Sources

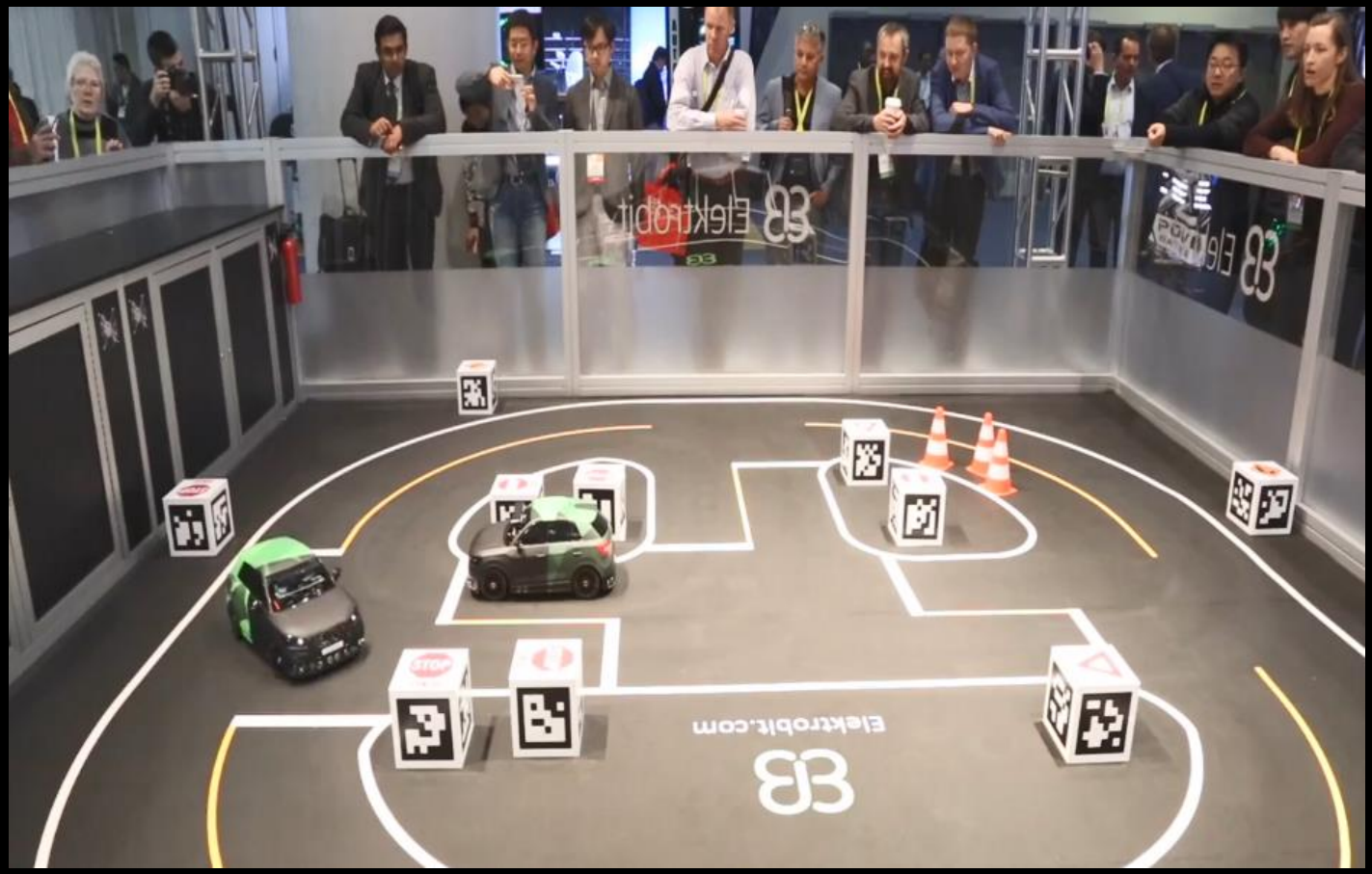
- **Complementary:** information provided by the input sources represents different parts of the scene and could be used to obtain more complete global information.
  - For example, in the case of visual sensor networks, the information on the same target provided by two cameras with different fields of view is considered complementary;
- **Redundant:** two or more input sources provide information about the same target and could thus be fused to increment the confidence.
  - For example, the data coming from overlapped areas in visual sensor networks are considered redundant;
- **Coperative:** the provided information is combined into new information that is typically more complex than the original information.
  - For example, multi-modal (audio and video) data fusion is considered cooperative.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

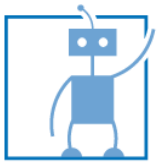
Classification Based on the Relations between the Data Sources



A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



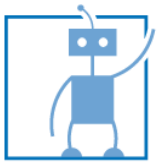
Impressions - CES 2017



Dasarathy's Classification

- (1) data in-data out (DAI-DAO): Data fusion at this level is conducted immediately after the data are gathered from the sensors. The algorithms employed at this level are based on signal and image processing algorithms.
- (2) data in-feature out (DAI-FEO): the data fusion process employs raw data from the sources to extract features or characteristics that describe an entity in the environment.
- (3) feature in-feature out (FEI-FEO): the data fusion process addresses a set of features with to improve, refine or obtain new features. This process is also known as feature fusion, symbolic fusion, information fusion or intermediatelevel fusion.
- (4) feature in-decision out (FEI-DEO): this level obtains a set of features as input and provides a set of decisions as output. Most of the classification systems that perform a decision based on a sensor's inputs fall into this category of classification;
- (5) Decision In-Decision Out (DEI-DEO): This type of classification is also known as decision fusion. It fuses input decisions to obtain better or new decisions.

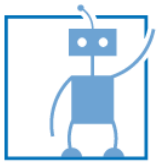
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Classification Based on the Abstraction Levels

- (1) signal level: directly addresses the signals that are acquired from the sensors;
- (2) pixel level: operates at the image level and could be used to improve image processing tasks;
- (3) characteristic: employs features that are extracted from the images or signals (i.e., shape or velocity),
- (4) symbol: at this level, information is represented as symbols; this level is also known as the decision level

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Classification Based on the Abstraction Levels

Alternatively:

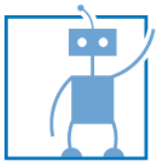
- (1) low level fusion: the raw data are directly provided as an input to the data fusion process, which provide more accurate data (a lower signal-to-noise ratio) than the individual sources.

  (2) medium level fusion: characteristics or features (shape, texture, and position) are fused to obtain features that could be employed for other tasks. This level is also known as the feature or characteristic level;

- (3) high level fusion: this level, which is also known as decision fusion, takes symbolic representations as sources and combines them to obtain a more accurate decision. Bayesian's methods are typically employed at this level;

- (4) multiple level fusion: this level addresses data provided from different levels of abstraction (i.e., when a measurement is combined with a feature to obtain a decision)

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



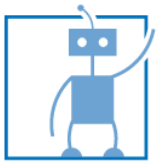
Classification Based on JDL / DoD

(i) sources: the sources are in charge of providing the input data. Different types of sources can be employed, such as sensors, a priori information (references or geographic data), databases, and human inputs.

(ii) human-computer interaction (HCI): HCI is an interface that allows inputs to the system from the operators and produces outputs to the operators. HCI includes queries, commands, and information on the obtained results and alarms.

(iii) database management system: the database management system stores the provided information and the fused results. This system is a critical component because of the large amount of highly diverse information that is stored.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Classification Based on JDL / DoD

(1) level 0—source preprocessing: source preprocessing is the lowest level of the data fusion process, and it includes fusion at the signal and pixel levels. In the case of text sources, this level also includes the information extraction process. This level reduces the amount of data and maintains useful information for the high-level processes.

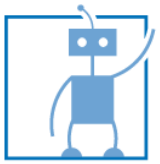
(2) level 1—object refinement: object refinement employs the processed data from the previous level. Common procedures of this level include spatio-temporal alignment, association, correlation, clustering or grouping techniques, state estimation, the removal of
false positives, identity fusion, and the combining of features that were extracted from images. The output results of this stage are the object discrimination (classification and identification) and object tracking (state of the object and orientation). This stage transforms the input information into consistent data structures.

(3) level 2—situation assessment: this level focuses on a higher level of inference than level 1. Situation assessment aims to identify the likely situations given the observed events and obtained data. It establishes relationships between the objects. Relations (i.e., proximity, communication) are valued to determine the significance of the entities or objects in a specific environment. The aim of this level includes performing high-level inferences and identifying significant activities and events (patterns in general). The output is a set of high-level inferences.

(4) level 3—impact assessment: this level evaluates the impact of the detected activities in level 2 to obtain a proper perspective. The current situation is evaluated, and a future projection is performed to identify possible risks, vulnerabilities, and operational opportunities. This level includes (1) an evaluation of the risk or threat and (2) a prediction of the logical outcome;

(5) level 4—process refinement: this level improves the process from level 0 to level 3 and provides resource and sensor management. The aim is to achieve efficient resource management while accounting for task priorities, scheduling, and the control of available resources.

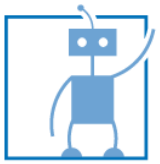
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Classification Based on the Type of Architecture

- Centralized Architecture
- Decentralized Architecture
- Distributed Architecture
- Hierarchical Architecture

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Centralized Architecture

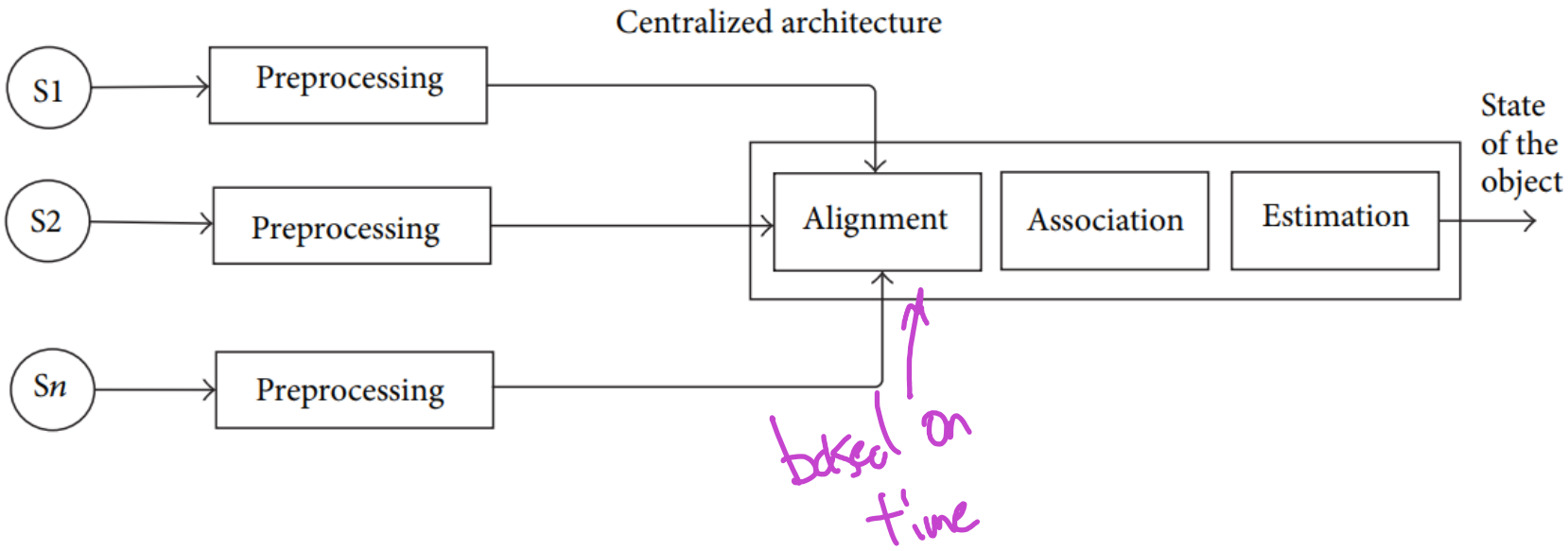
- in a centralized architecture, the fusion node resides in the central processor that receives the information from all of the input sources. All of the fusion processes are executed in a central processor that uses the provided raw measurements from the sources.

- The sources obtain only the observations as measurements and transmit them to a central processor, where the data fusion process is performed.

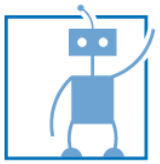
- If we assume that data alignment and data association are performed correctly and that the required time to transfer the data is not significant, then the centralized scheme is theoretically optimal. However, the assumptions typically do not hold for real systems.
  - The large amount of bandwidth that is required to send raw data through the network is a disadvantage for the centralized approach.
  - This issue becomes a bottleneck when this type of architecture is employed for fusing data in visual sensor networks.
  - Finally, the time delays when transferring the information between the different sources are variable and affect the results in the centralized scheme to a greater degree than in other schemes;

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

Centralized Architecture



A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

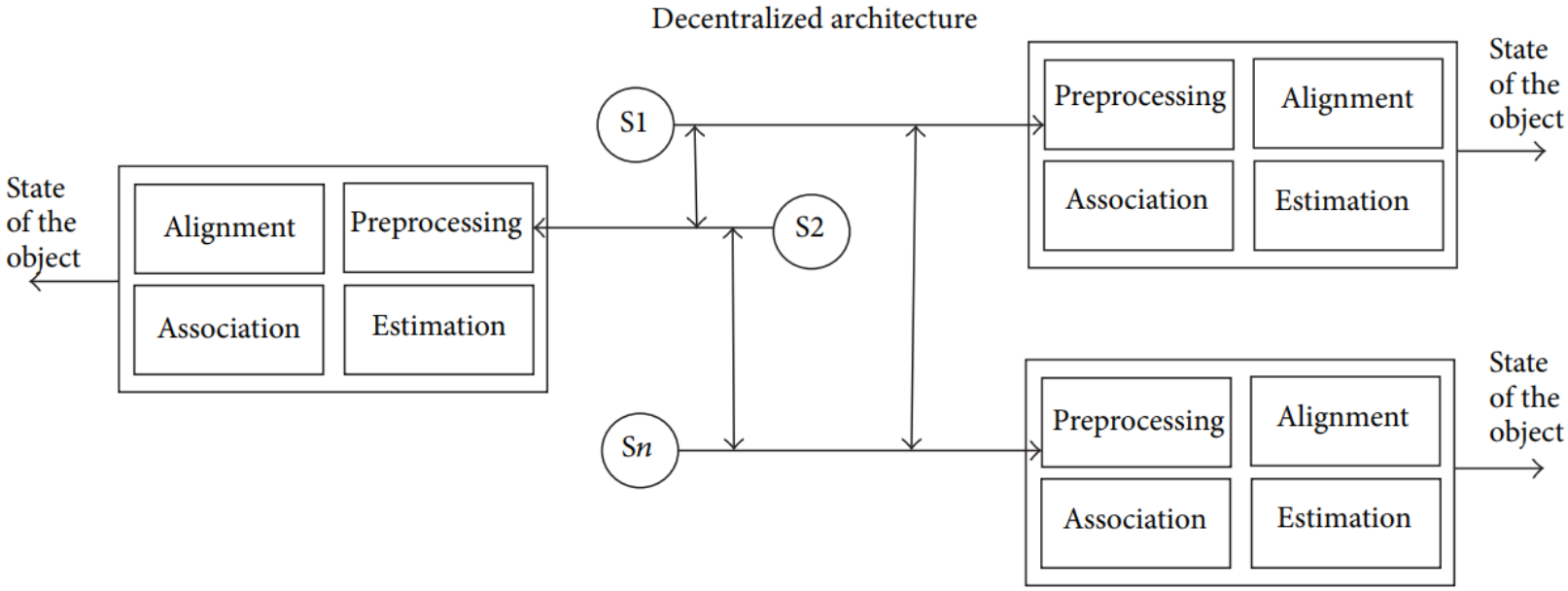


Decentralized Architecture

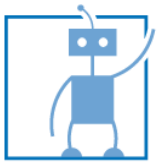
- Composed of a network of nodes in which each node has its own processing capabilities and there is no single point of data fusion.
- Each node fuses its local information with the information that is received from its peers. Data fusion is performed autonomously, with each node accounting for its local information and the information received from its peers.
- The main disadvantage of this architecture is the communication cost, which is $O(n^2)$ at each communication step, where n is the number of nodes additionally, the extreme case is considered, in which each node communicates with all of its peers.
- Thus, this type of architecture could suffer from scalability problems when the number of nodes is increased. Alternatively fused information could not be available where it is needed.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

Decentralized Architecture



A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

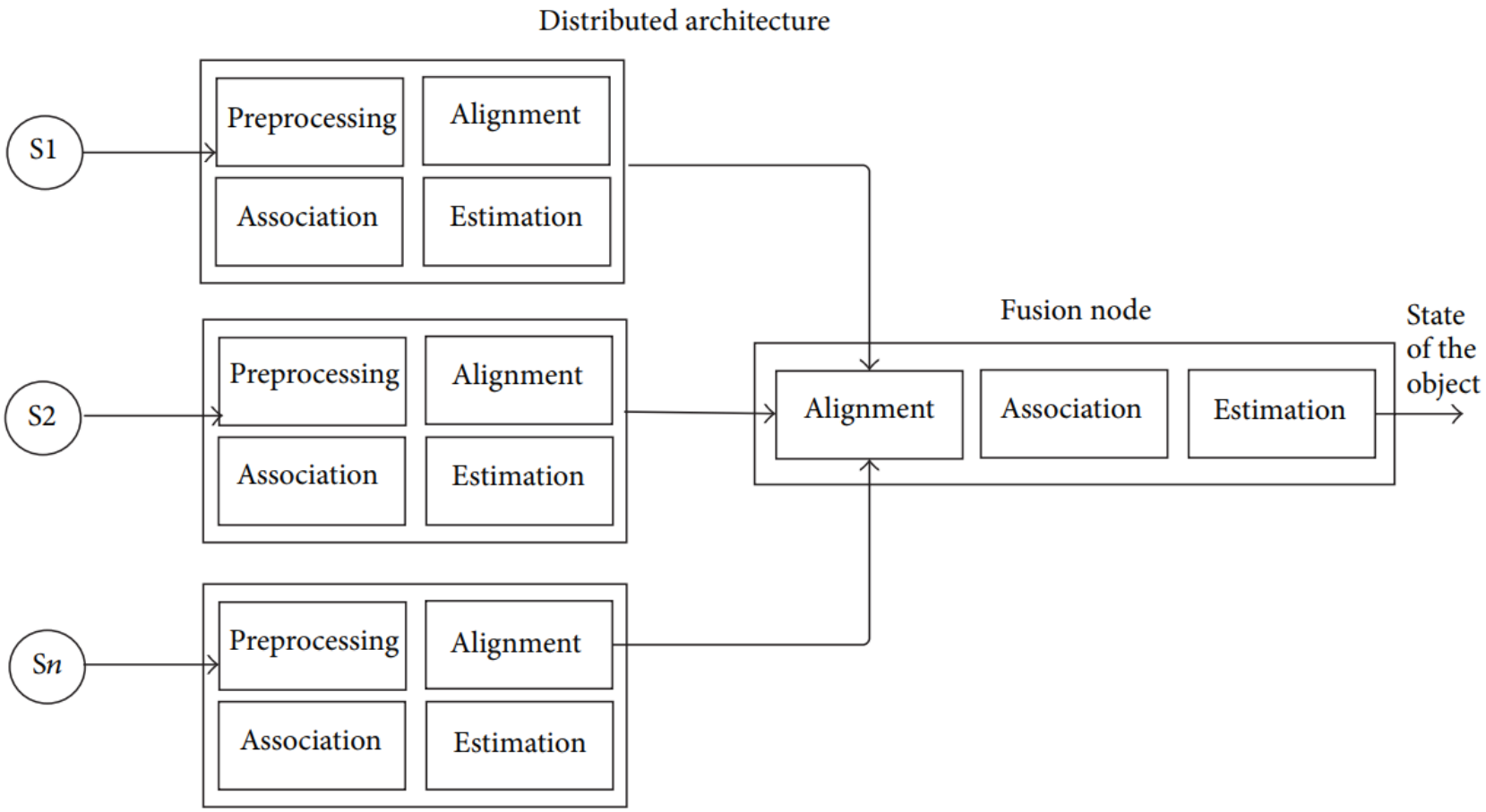


Distributed Architecture

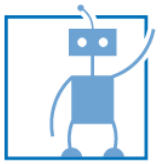
- In a distributed architecture, measurements from each source node are processed independently before the information is sent to the fusion node.
- The fusion node accounts for the information that is received from the other nodes. In other words, the data association and state estimation are performed in the source node before the information is communicated to the fusion node.
- Therefore, each node provides an estimation of the object state based on only their local views, and this information is the input to the fusion process, which provides a fused global view.
- This type of architecture provides different options and variations that range from only one fusion node to several intermediate fusion nodes.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

Distributed Architecture



A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



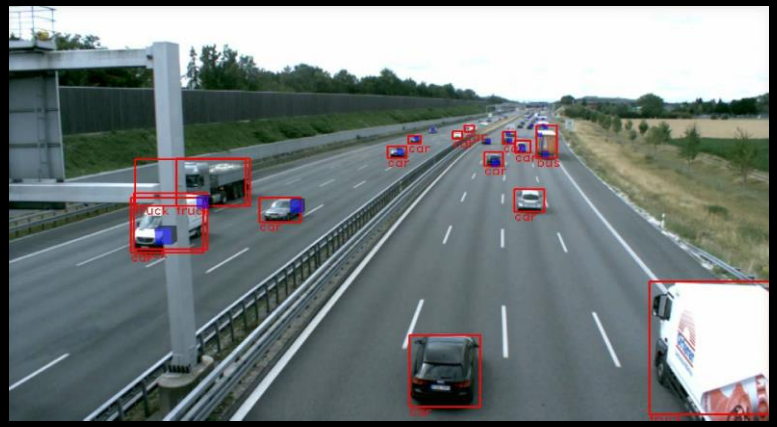
Hierarchical Architecture

- Hierarchical architectures often comprise a combination of decentralized and distributed nodes, generating hierarchical schemes in which the data fusion process is performed at different levels in the hierarchy.

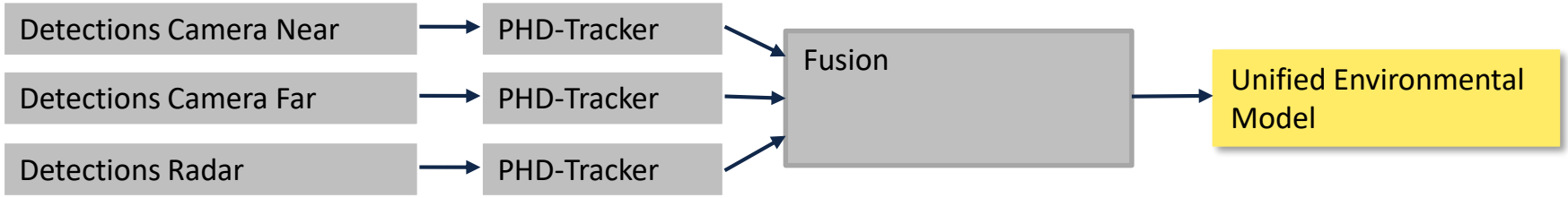
Conclusion:

in practice, there is no single best architecture, and the selection of the most appropriate architecture should be made depending on the requirements, demand, existing networks, data availability, node processing capabilities, and organization of the data fusion system.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Flexible Multisensor Data Fusion with distributed, adaptive, autocalibrated sensors

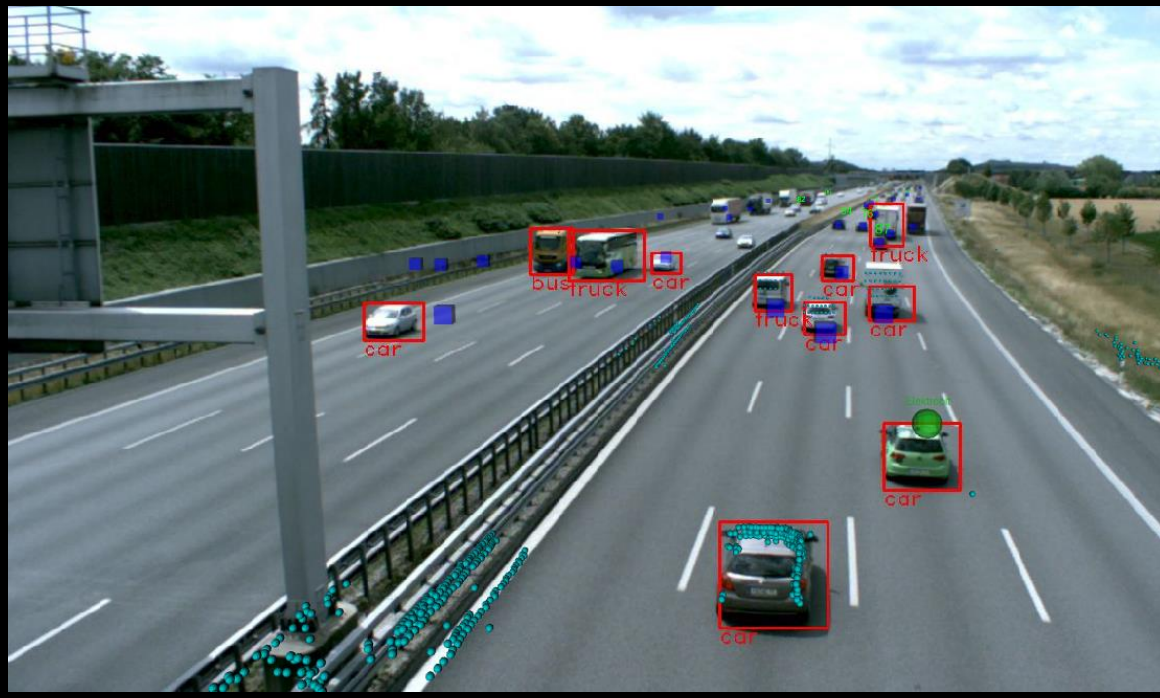


C2X Sensors

- Connected system with shared sensor information

- **Infrastructure**
  - Radar
  - Camera Near
  - Camera Far 123

- **Fahrzeug:**
  - Ego-vehicle detection
  - Lidar

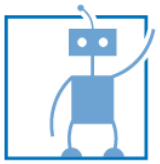


MP South

MP North

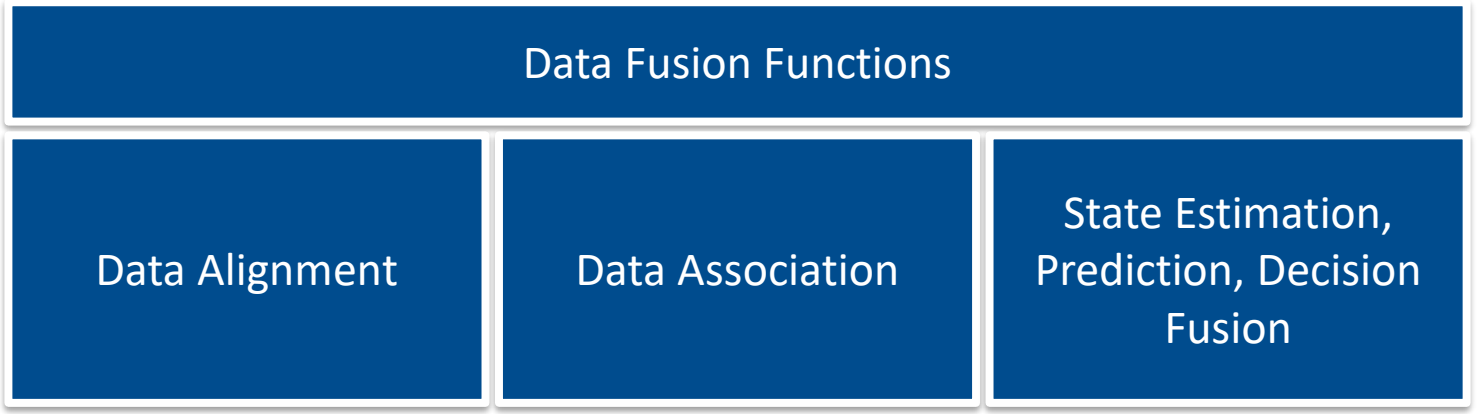
Near

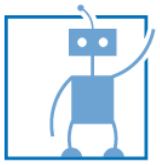
Far



Data Fusion Functions

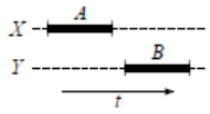
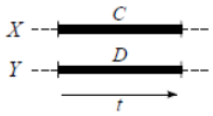
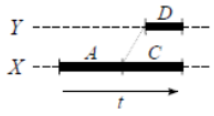
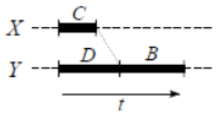
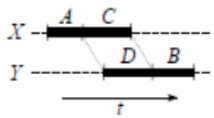
- A number of sub tasks are of high relevance for MSDF.
- By splitting MSDF into these subtasks, they can be studied individually
- Data Fusion functions include: Data Alignment, Data Association, State estimation, prediction, decision fusion



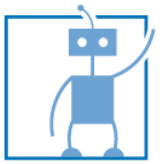


Data Alignment

- Detecting temporal differences, references and adjusting the collected data according to the detected references.
- Synchronize using shared time.
- Synchronize via communication protocol.
- Detect patterns, such as concurrency, among the generated streams.
- Spatio-temporal, data normalization, evidence conditioning.

Logical mappings		Scenario example
precedes:	$A \rightarrow B$	
simultaneous:	$C \mid\mid\mid D$	
ends:	$A \rightarrow (C \mid\mid\mid D)$	
starts:	$(C \mid\mid\mid D) \rightarrow B$	
overlaps:	$A \rightarrow (C \mid\mid\mid D) \rightarrow B$	

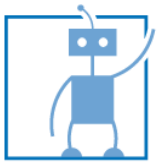
Data Alignment for Data Fusion in Wireless Multimedia Sensor Networks Based on M2M, Jose Roberto Perez Cruz et al., KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 6, NO. 1, Jan 2012



Data Association

- The data association problem must determine the set of measurements that correspond to each target.
- Suppose that there are O targets that are being tracked by only one sensor in a cluttered environment (by a cluttered environment, we refer to an environment that has several targets that are to close each other). Then, the data association problem can be defined as follows:
- (i) each sensor's observation is received in the fusion node at discrete time intervals
- (ii) the sensor might not provide observations at a specific interval
- (iii) some observations are noise, and other observations originate from the detected target
- (iv) for any specific target and in every time interval, we do not know (a priori) the observations that will be generated by that target
- The goal of data association is to establish the set of observations or measurements that are generated by the same target over time.
- Hall and Llinas: "The process of assigning and computing the weights that relates the observations or tracks (A track can be defined as an ordered set of points that follow a path and are generated by the same target.) from one set to the observation of tracks of another set."

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



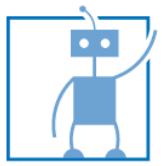
Data Association

- Illustration of the complexity of the data association problem:
  - if we take a frame-to-frame association and assume that M possible points could be detected in all n frames, then the number of possible sets is $(M!)^{n-1}$. Note that from all of these possible solutions, only one set establishes the true movement of the M points.

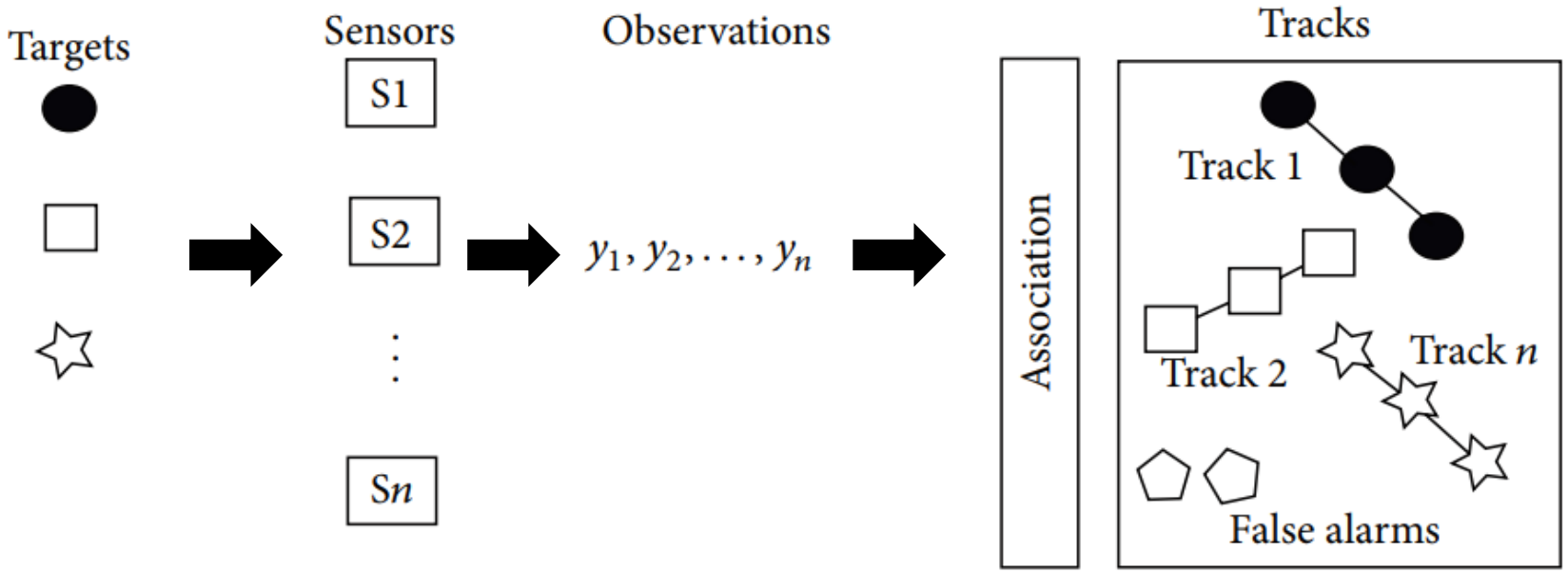
- Data association is often performed before the state estimation of the detected targets. Moreover, it is a key step because the estimation or classification will behave incorrectly if the data association phase does not work coherently. The data association process could also appear in all of the fusion levels, but the granularity varies depending on the objective of each level.

- In general, an exhaustive search of all possible combinations grows exponentially with the number of targets. The data association problem becomes NP complete.

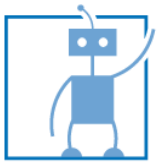
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Data Association



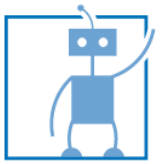
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Data Association

- Examples of Data association methods:
  - Clustering: Nearest Neighbors and K-Means, Gaussian Mixture Models (GMM).
  - Nearest neighbor (NN) is the simplest data association technique. NN is a well-known clustering algorithm that selects or groups the most similar values. How close the one measurement is to another depends on the employed distance metric and typically depends on the threshold that is established by the designer. In general, the employed criteria could be based on (1) an absolute distance, (2) the Euclidean distance, or (3) a statistical function of the distance
  - NN is a simple algorithm that can find a feasible (approximate) solution in a small amount of time. However, in a cluttered environment, it could provide many pairs that have the same probability and could thus produce undesirable error propagation. Moreover, this algorithm has poor performance in environments in which false measurements are frequent, which are in highly noisy environments. All neighbors use a similar technique, in which all of the measurements inside a region are included in the tracks.

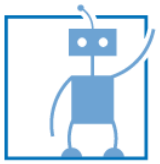
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Data Association

- Examples of Data association methods:
  - Clustering: Nearest Neighbors and K-Means, Gaussian Mixture Models (GMM).
  - K-Means method is a well-known modification of the NN algorithm. K-Means divides the dataset values into K different clusters. K-Means algorithm finds the best localization of the cluster centroids, where best means a centroid that is in the center of the data cluster. K-Means is an iterative algorithm that can be divided into the following steps:
  - K-Means is a popular algorithm that has been widely employed; however, it has the following disadvantages:
    - (i) the algorithm does not always find the optimal solution for the cluster centers;
    - (ii) the number of clusters must be known a priori and one must assume that this number is the optimum;
    - (iii) the algorithm assumes that the covariance of the dataset is irrelevant or that it has been normalized already.
  - There are several options for overcoming these limitations. For the first one, it is possible to execute the algorithm several times and obtain the solution that has less variance. For the second one, it is possible to start with a low value of K and increment the values of K until an adequate result is obtained. The third limitation can be easily overcome by multiplying the data with the inverse of the covariance matrix

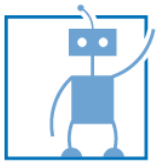
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Data Association

- Examples of Data association methods:
  - Probabilistic Data Association
  - The probabilistic data association (PDA) algorithm was proposed by Bar-Shalom and Tse and is also known as the modified filter of all neighbors. This algorithm assigns an association probability to each hypothesis from a valid measurement of a target.
  - A valid measurement refers to the observation that falls in the validation gate of the target at that time instant.
  - The validation gate, γ, which is the center around the predicted measurements of the target, is used to select the set of basic measurements
  - In the PDA algorithm, the state estimation of the target is computed as a weighted sum of the estimated state under all of the hypotheses. The algorithm can associate different measurements to one specific target. Thus, the association of the different measurements to a specific target helps PDA to estimate the target state, and the association probabilities are used as weights.

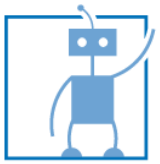
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



Data Association

- Examples of Data association methods:
  - Probabilistic Data Association
  - The main disadvantages of the PDA are:
  - (i) loss of tracks: because PDA ignores the interference with other targets, it sometimes could wrongly classify the closest tracks. Therefore, it provides a poor performance when the targets are close to each other or crossed.
  - (ii) the suboptimal Bayesian approximation: when the source of information is uncertain, PDA is the suboptimal Bayesian approximation to the association problem.
  - (iii) one target: PDA was initially designed for the association of one target in a low-cluttered environment. The number of false alarms is typically modeled with the Poisson distribution, and they are assumed to be distributed uniformly in space. PDA behaves incorrectly when there are multiple targets because the false alarm model does not work well.
  - (iv) track management: because PDA assumes that the track is already established, algorithms must be provided for track initialization and track deletion.
  - PDA is mainly good for tracking targets that do not make abrupt changes in their movement patterns. PDA will most likely lose the target if it makes abrupt changes in its movement patterns.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



State Estimation Methods

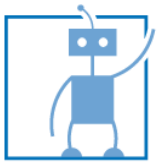
- State estimation techniques aim to determine the state of the target under movement (typically the position) given State estimation techniques aim to determine the state of the target under movement (typically the position) given the observation or measurements.

- State estimation techniques are also known as tracking techniques. In their general form, it is not guaranteed that the target observations are relevant, which means that some of the observations could actually come from the target and others could be only noise.

- The state estimation phase is a common stage in data fusion algorithms because the target's observation could come from different sensors or sources, and the final goal is to obtain a global target state from the observations.

- The estimation problem involves finding the values of the vector state (e.g., position, velocity, and size) that fits as much as possible with the observed data. From a mathematical perspective, we have a set of redundant observations, and the goal is to find the set of parameters that provides the best fit to the observed data. In general, these observations are corrupted by errors and the propagation of noise in the measurement process.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



State Estimation Methods

- State estimation methods fall under level 1 of the JDL classification and could be divided into two broader groups:
  - (1) linear dynamics and measurements: here, the estimation problem has a standard solution. Specifically, when the equations of the object state and the measurements are linear, the noise follows the Gaussian distribution, and we do not refer to it as a clutter environment; in this case, the optimal theoretical solution is based on the Kalman filter;
  - (2) nonlinear dynamics: the state estimation problem becomes difficult, and there is not an analytical solution to solve the problem in a general manner. In principle, there are no practical algorithms available to solve this problem satisfactorily

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

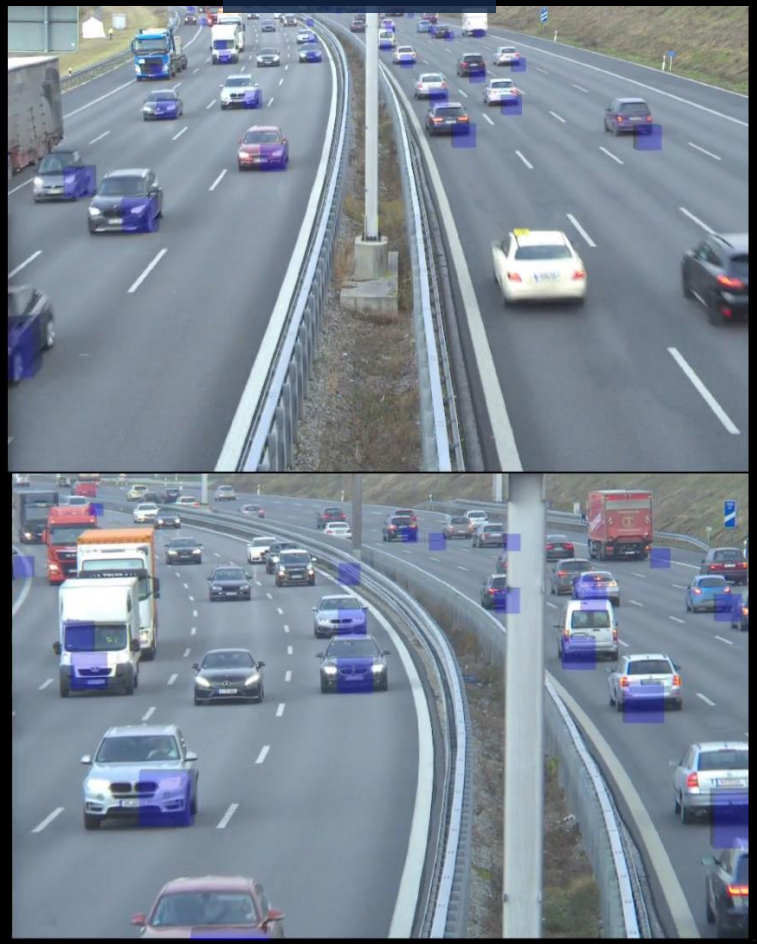
With component you mean by linear? lane, car?

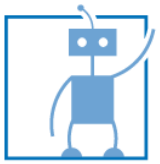
MP South

MP North

Near

Far





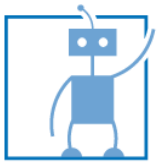
State Estimation Methods

- State estimation methods fall under level 1 of the JDL classification and could be divided into two broader groups:

  - Most of the state estimation methods are based on control theory and employ the laws of probability to compute a vector state from a vector measurement or a stream of vector measurements.

  - Common estimation methods include:
  - maximum likelihood and maximum posterior
  - Kalman filter importend
  - particle filter
  - Distributed Kalman filter
  - Distributed particle filter
  - covariance consistency methods
  - …

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



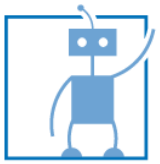
State Estimation Methods

Maximum likelihood and maximum posterior

- The maximum likelihood (ML) technique is an estimation method that is based on probabilistic theory.
- Probabilistic estimation methods are appropriate when the state variable follows an unknown probability distribution.
- In the context of data fusion, x is the state that is being estimated, and $z = (z(1), \ldots, z(k))$ is a sequence of k previous observations of x.
- Likelihood function $\lambda(x)$ is defined as a probability density function of the sequence of z observations given the true value of the state x.
  $\lambda(x) = p(z \mid x)$.
- The ML estimator finds the value of x that maximizes the likelihood function, which can be obtained from the analytical or empirical models of the sensors.

$$\hat{x}(k) = \arg\max_{x} p(z \mid x)$$

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages



State Estimation Methods

Maximum likelihood and maximum posterior

- The ML estimator finds the value of x that maximizes the likelihood function, which can be obtained from the analytical or empirical models of the sensors.

$$\hat{x}(k) = \arg\max_{x} p(z \mid x)$$

- This function expresses the probability of the observed data. The main disadvantage of this method in practice is that it requires the analytical or empirical model of the sensor to be known to provide the prior distribution and compute the likelihood function.

- This method can also systematically underestimate the variance of the distribution, which leads to a bias problem. However, the bias of the ML solution becomes less significant as the number N of data points increases and is equal to the true variance of the distribution that generated the data at the limit $N \to \infty$

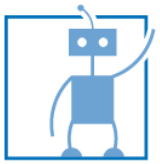
A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

MP South

MP North

Near

Far



State Estimation Methods

Maximum likelihood and maximum posterior

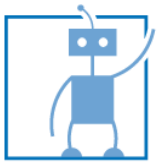
- The maximum posterior (MAP) method is based on the Bayesian theory. It is employed when the parameter x to be estimated is the output of a random variable that has a known probability density function $p(x)$. In the context of data fusion, x is the state that is being estimated and $z = (z(1), \ldots, z(k))$ is a sequence of k previous observations of x.

- The MAP estimator finds the value of x that maximizes the posterior probability distribution as follows:

$$\hat{x}(k) = \arg\max_{x} p(x \mid z)$$

- Both methods (ML and MAP) aim to find the most likely value for the state x. However, ML assumes that x is a fixed but an unknown point from the parameter space, whereas MAP considers x to be the output of a random variable with a known a priori probability density function. Both of these methods are equivalent when there is no a priori information about x, that is, when there are only observations.

A Review of Data Fusion Techniques, Federico Castanedo, Hindawi Publishing Corporation
The Scientific World Journal, Volume 2013, Article ID 704504, 19 pages

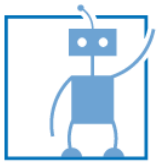


State Estimation Methods

Kalman Filter

Example: Estimate orientation from measured velocities

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

Kalman filter is useful to deal with uncertain information about a dynamic system, when a prediction can be made about its behavior.

Advantage that they are light on memory (they don't need to keep any history other than the previous state), and they are very fast, making them well suited for real time problems and embedded systems.

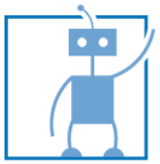
Introduction (after Bzarg phi, Kalman Filter introduction. Can check for extended explanation):

$\overrightarrow{x_k}$: robot state, $\vec{p}$: robot position, $\vec{v}$: robot velocity

$$\overrightarrow{x_k} = (\vec{p}, \vec{v})$$

Extended watching: https://classroom.udacity.com/courses/cs373/ or S. Thrun

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



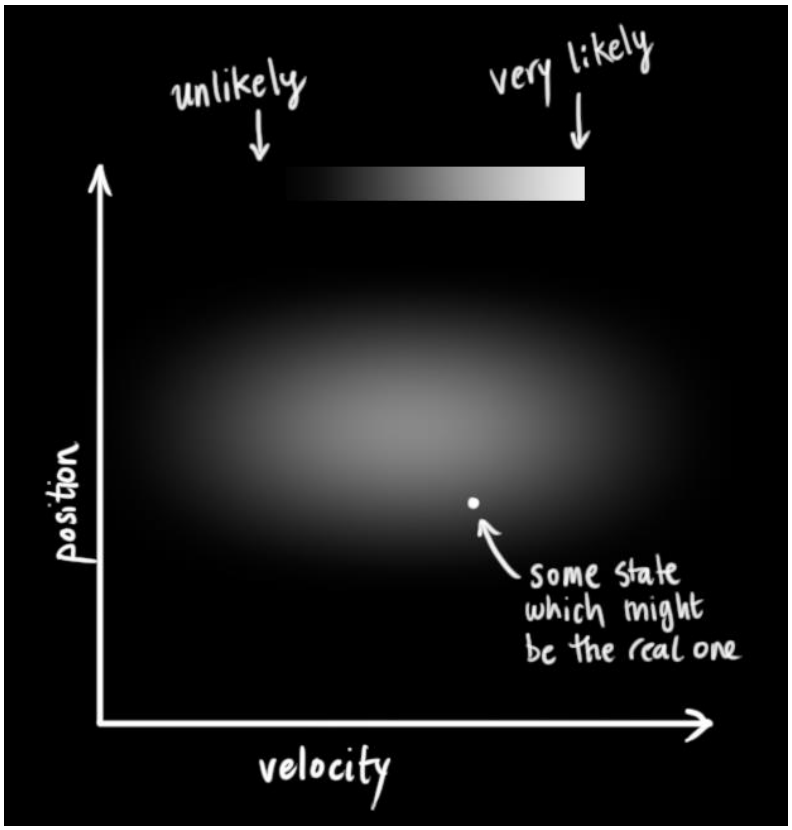
State Estimation Methods

Kalman Filter

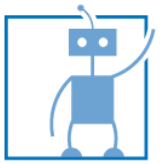
$\overrightarrow{\hat{x}_k}$: robot state, $\vec{p}$: robot position, $\vec{v}$: robot velocity

$$\overrightarrow{x_k} = (\vec{p}, \vec{v})$$

$$\vec{x} = \begin{bmatrix} p \\ v \end{bmatrix}$$



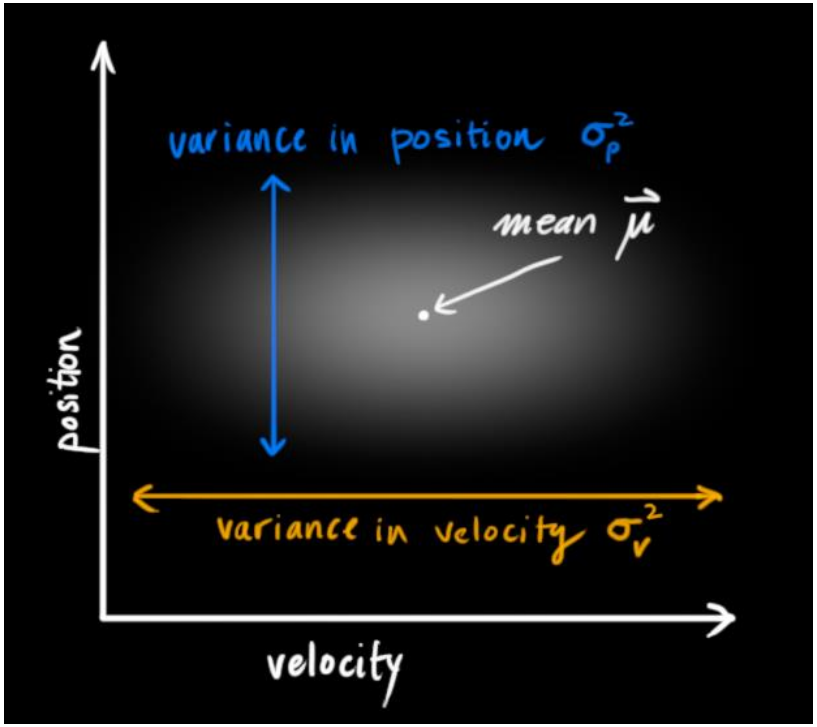
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

Kalman filter assumes both state variables (position and velocity, in our case) are random and Gaussian distributed. Each variable has a mean value μ, which is the center of the random distribution (and its most likely state), and a variance σ^2.

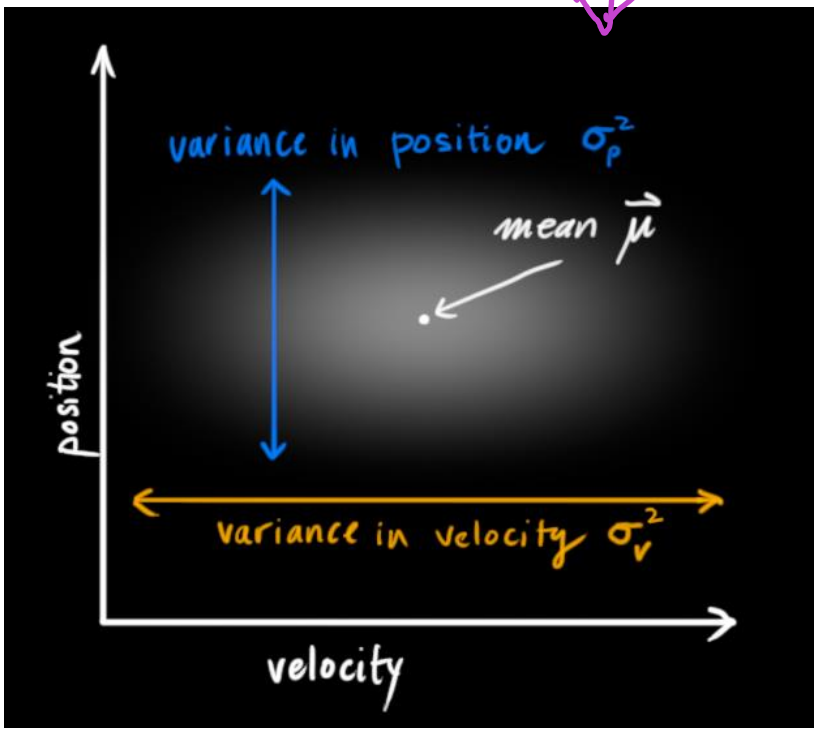


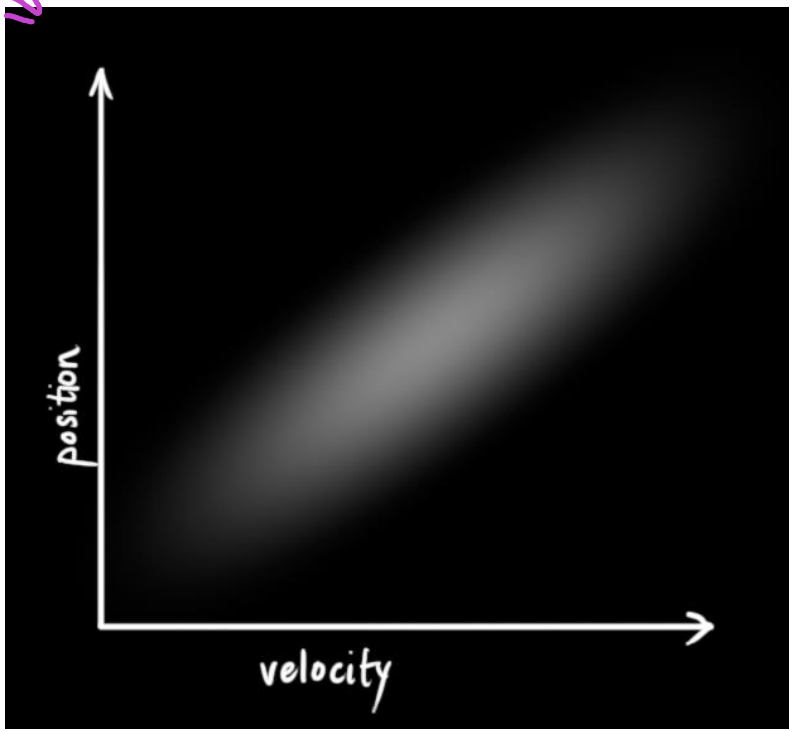
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

State Estimation Methods

Kalman Filter

- Left: position and velocity are **uncorrelated. S**tate of one variable tells nothing the other.
- Right: Position and velocity are **correlated**. The likelihood of observing a particular position depends on the velocity.



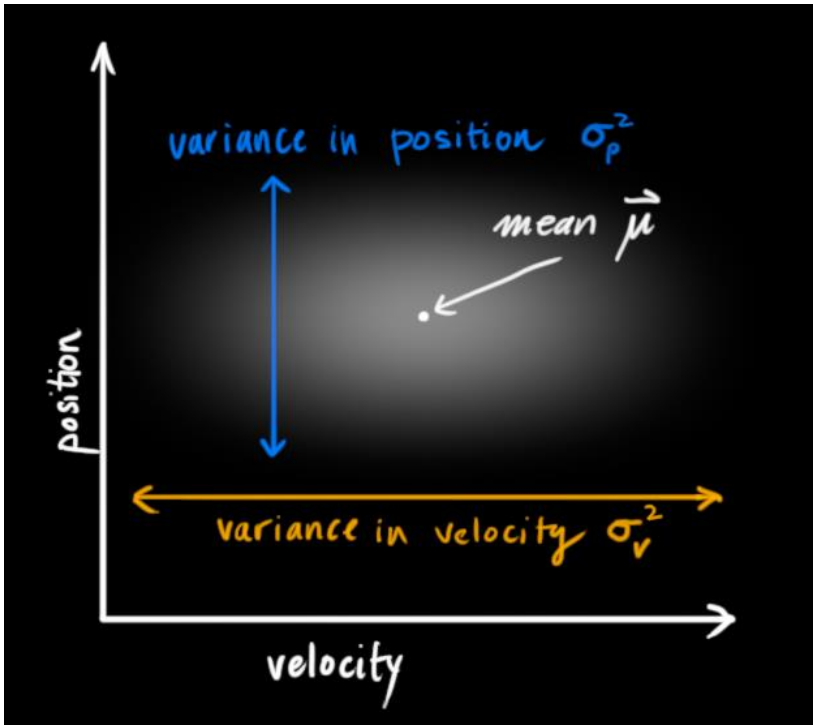


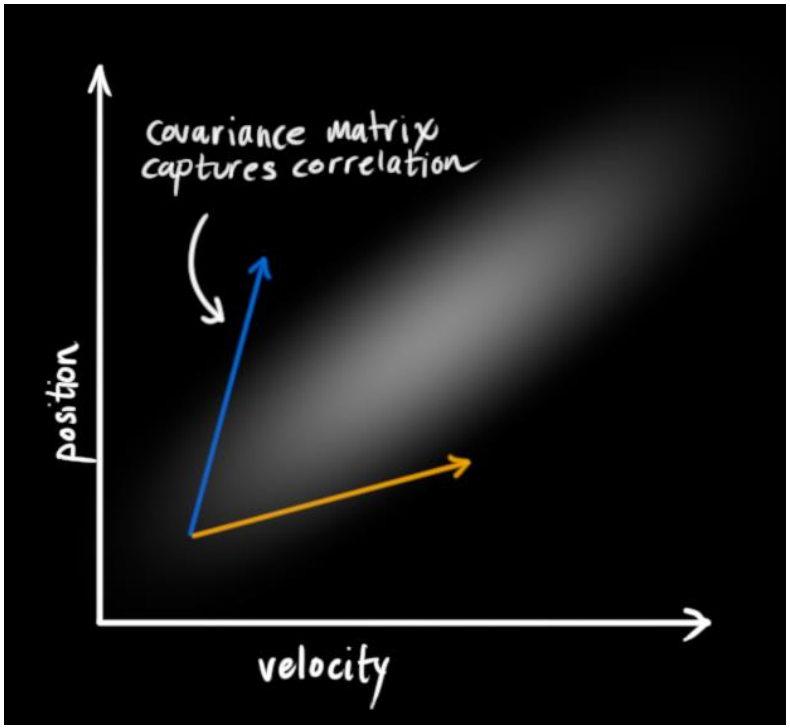
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

State Estimation Methods

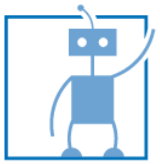
Kalman Filter

- Correlation is captured by a covariance matrix. Each element of the matrix Σ_{ij} is the degree of correlation between the ith state variable and the jth state variable. The covariance matrix is symmetric.





Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- Correlation is captured by a covariance matrix. Each element of the matrix Σ_{ij} is the degree of correlation between the ith state variable and the jth state variable. The covariance matrix is symmetric.
- $\overrightarrow{\hat{x}_k}$: robot state estimate
- $\mathbf{P}_k$: covariance matrix

$$\hat{\mathbf{x}}_k = \begin{bmatrix} \text{position} \\ \text{velocity} \end{bmatrix}$$

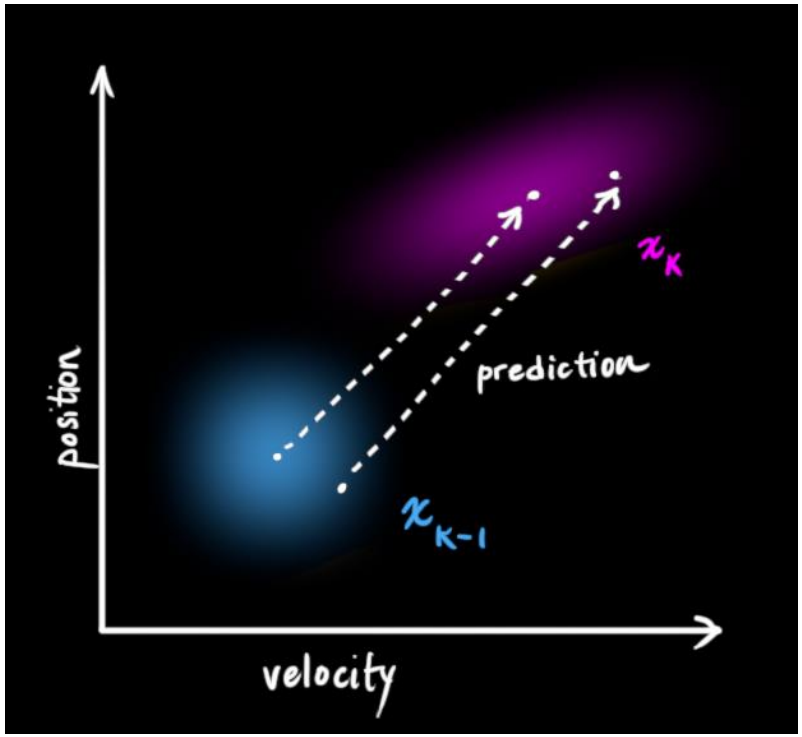
$$\mathbf{P}_k = \begin{bmatrix} \Sigma_{pp} & \Sigma_{pv} \\ \Sigma_{vp} & \Sigma_{vv} \end{bmatrix}$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

State Estimation Methods

Kalman Filter

- Prediction: Look at current state (at time k-1) and predict the next state at time k. Compute new distribution.

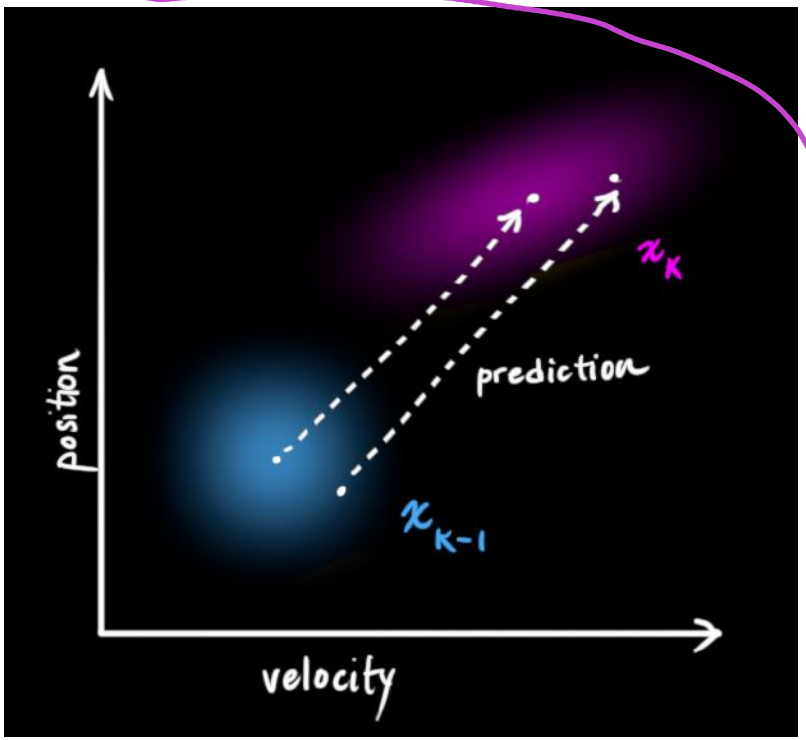


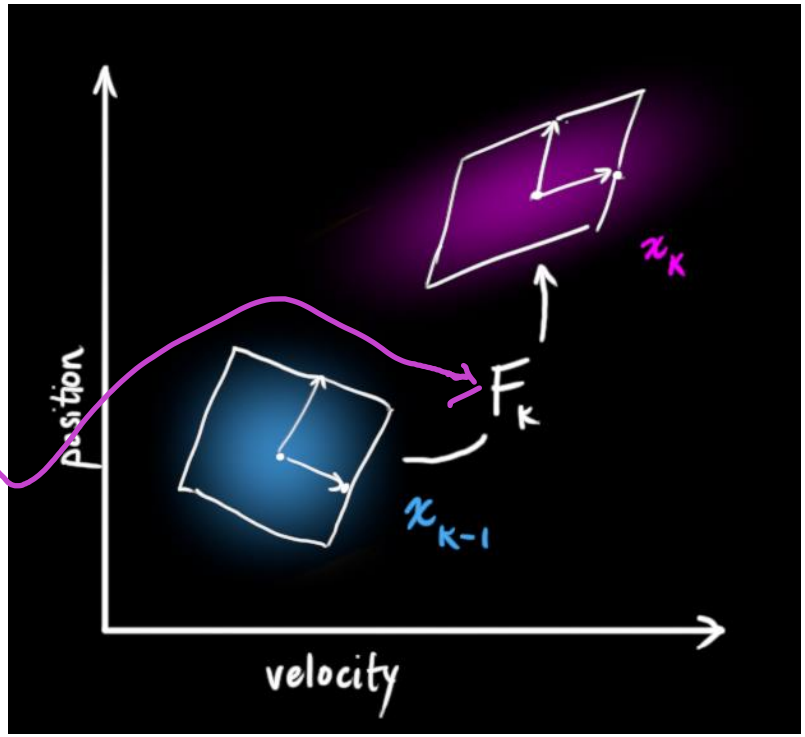
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

State Estimation Methods

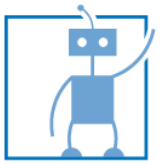
Kalman Filter

- Prediction: Look at current state (at time k-1) and predict the next state at time k. Compute new distribution.
- Transition matrix $\mathbf{F}_k$ between x_{k-1} and x_k.





Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

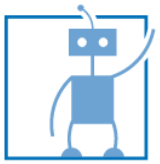
Kalman Filter

- Prediction: Look at current state (at time k-1) and predict the next state at time k. Compute new distribution.
- Transition matrix $\mathbf{F}_k$ between x_{k-1} and x_k.

$$\begin{aligned} p_k &= p_{k-1} + \Delta t v_{k-1} \\ v_k &= v_{k-1} \end{aligned}$$

$$\begin{aligned} \hat{\mathbf{x}}_k &= \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \hat{\mathbf{x}}_{k-1} \\ &= \mathbf{F}_k \hat{\mathbf{x}}_{k-1} \end{aligned}$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- Prediction: Look at current state (at time k-1) and predict the next state at time k. Compute new distribution.
- Transition matrix $\mathbf{F}_k$ between x_{k-1} and x_k.

Covariance

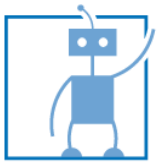
$$Cov(x) = \Sigma$$

$$Cov(\mathbf{A}x) = \mathbf{A}\Sigma\mathbf{A}^T$$

$$\hat{\mathbf{x}}_k = \mathbf{F}_k\hat{\mathbf{x}}_{k-1}$$

$$\mathbf{P}_k = \mathbf{F}_\mathbf{k}\mathbf{P}_{k-1}\mathbf{F}_k^T$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

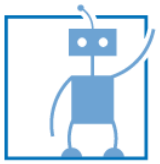
Kalman Filter

- Prediction: Look at current state (at time k-1) and predict the next state at time k. Compute new distribution.
- Transition matrix $\mathbf{F}_k$ between x_{k-1} and x_k.

$$Cov(x) = \Sigma \qquad \hat{\mathbf{x}}_k = \mathbf{F}_k\hat{\mathbf{x}}_{k-1}$$

$$Cov(\mathbf{A}x) = \mathbf{A}\Sigma\mathbf{A}^T \qquad \mathbf{P}_k = \mathbf{F}_\mathbf{k}\mathbf{P}_{k-1}\mathbf{F}_k^T$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- External influence: if the state models the motion of a vehicle, the train operator might push on the throttle, causing the vehicle to accelerate. If we know this additional information about what's going on in the world, we could include it into a control vector called $\overrightarrow{u_k}$ and add it to our prediction as a correction. $\mathbf{B}_k$: control matrix.

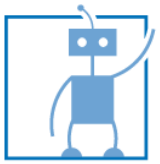
$$\begin{aligned} p_k &= p_{k-1} + \Delta t v_{k-1} \\ v_k &= v_{k-1} \end{aligned}$$

$$\begin{aligned} \hat{\mathbf{x}}_k &= \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \hat{\mathbf{x}}_{k-1} \\ &= \mathbf{F}_k \hat{\mathbf{x}}_{k-1} \end{aligned}$$

$$\begin{aligned} p_k &= p_{k-1} + \Delta t v_{k-1} + \frac{1}{2} a \Delta t^2 \\ v_k &= v_{k-1} + a \Delta t \end{aligned}$$

$$\begin{aligned} \hat{\mathbf{x}}_k &= \mathbf{F}_k \hat{\mathbf{x}}_{k-1} + \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{bmatrix} a \\ &= \mathbf{F}_k \hat{\mathbf{x}}_{k-1} + \mathbf{B}_k \overrightarrow{\mathbf{u}_k} \end{aligned}$$

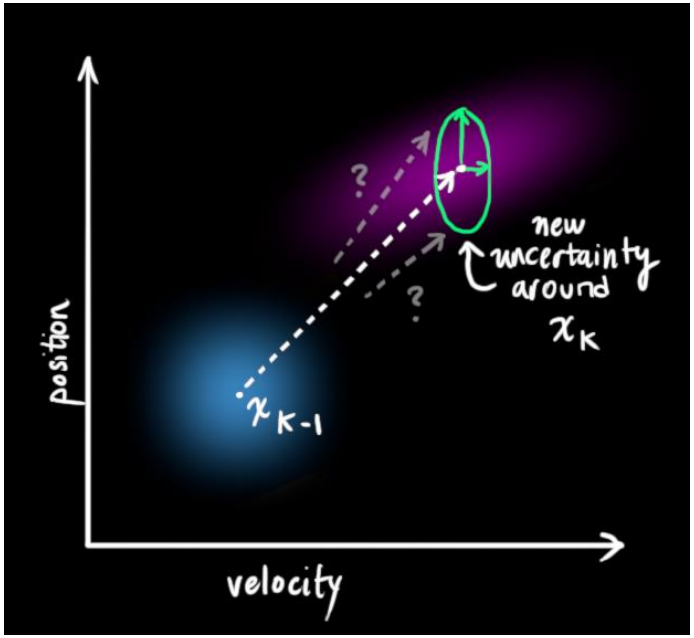
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

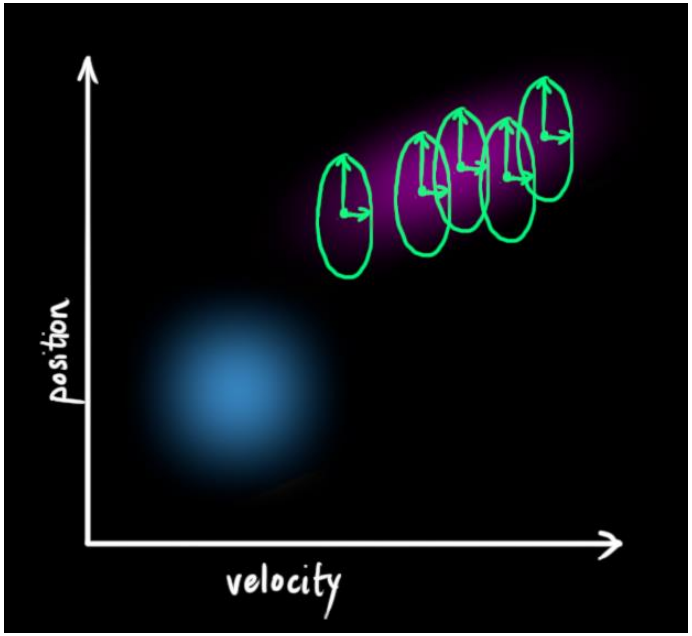


State Estimation Methods

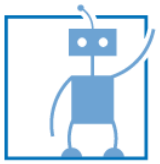
Kalman Filter

- Coping with prediction errors due to external unknown forces. Example: tracking a quadcopter, it could be buffeted around by wind. Modelled by adding some new uncertainty after every prediction step. Untracked influences are treated as noise with covariance Q_k.





Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- Prediction Step:

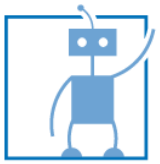
In other words, the **new best estimate** is a **prediction** made from **previous best estimate**, plus a **correction** for **known external influences**.

And the **new uncertainty** is **predicted** from the **old uncertainty**, with some **additional uncertainty from the environment**.

$$\hat{\mathbf{x}}_k = \mathbf{F}_k\hat{\mathbf{x}}_{k-1} + \mathbf{B}_k\overrightarrow{\mathbf{u}_k}$$

$$\mathbf{P}_k = \mathbf{F_k}\mathbf{P}_{k-1}\mathbf{F}_k^T + \mathbf{Q}_k$$

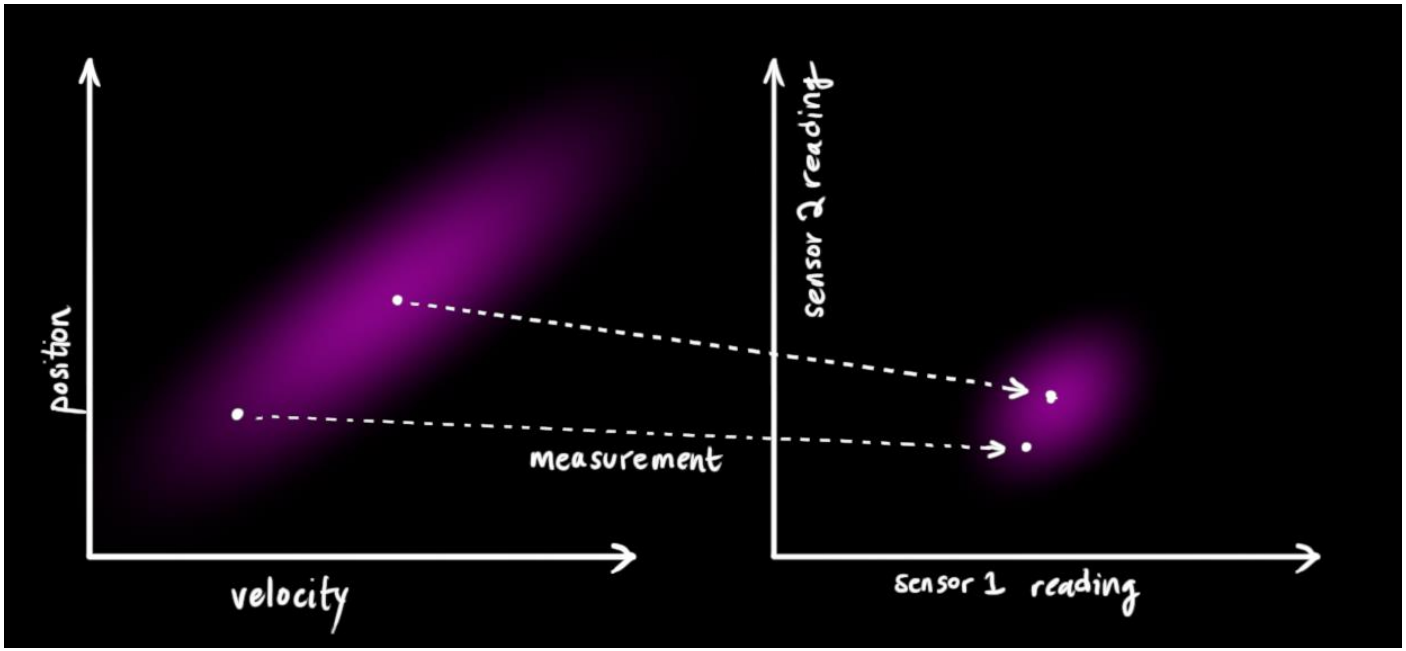
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



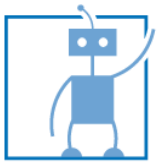
State Estimation Methods

Kalman Filter

- Including sensor measurement $\overrightarrow{z_k}$.



Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

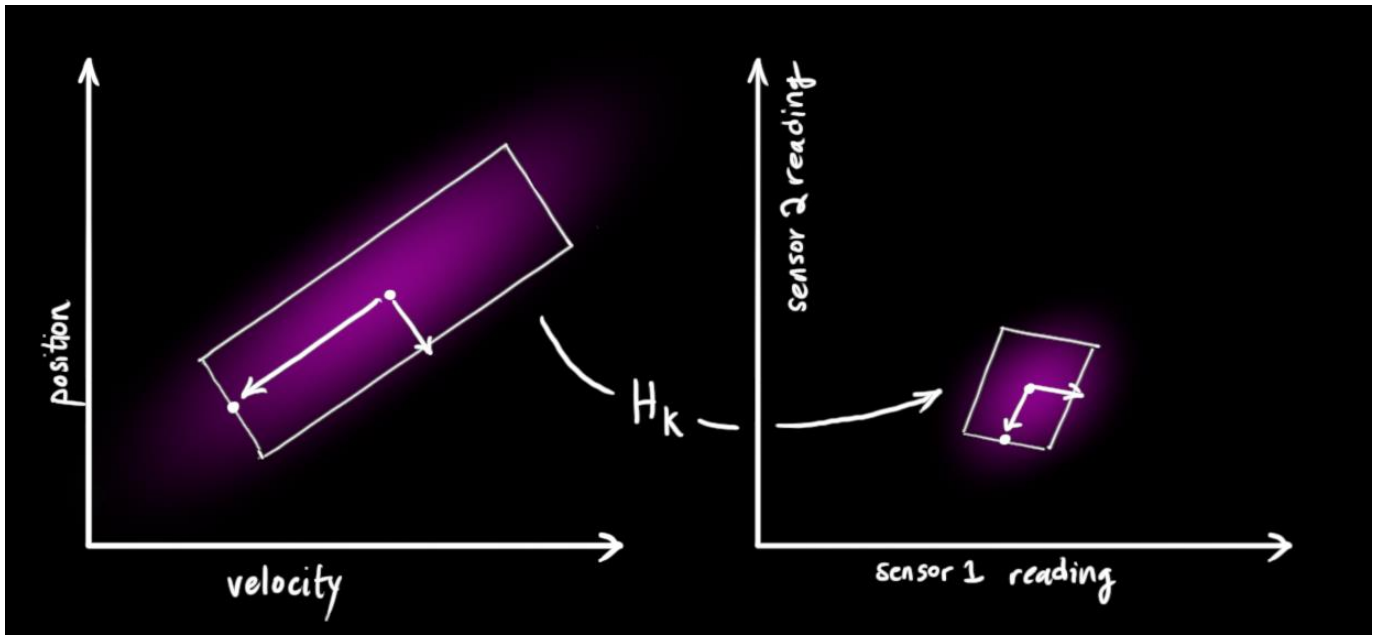


State Estimation Methods

Kalman Filter

- Including sensor measurement $\overrightarrow{z_k}$. Model expected sensor measurement with matrix $\mathbf{H}_k$.

$$\vec{\mu}_{\text{expected}} = \mathbf{H}_k \hat{\mathbf{x}}_k$$
$$\mathbf{\Sigma}_{\text{expected}} = \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T$$

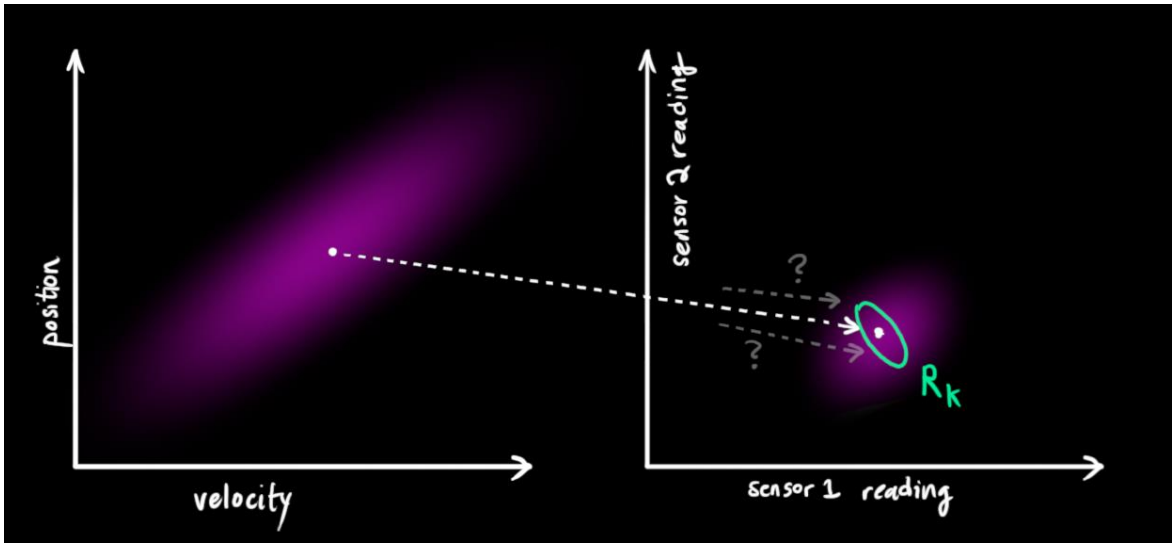


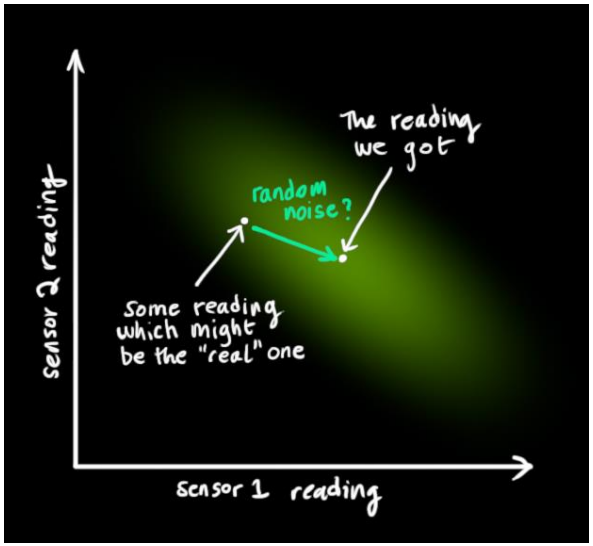
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

State Estimation Methods

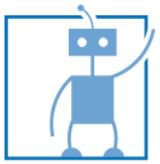
Kalman Filter

- Including sensor measurement $\overrightarrow{z_k}$. Model expected sensor measurement with matrix $\mathbf{H}_k$.
- Include sensor noise.





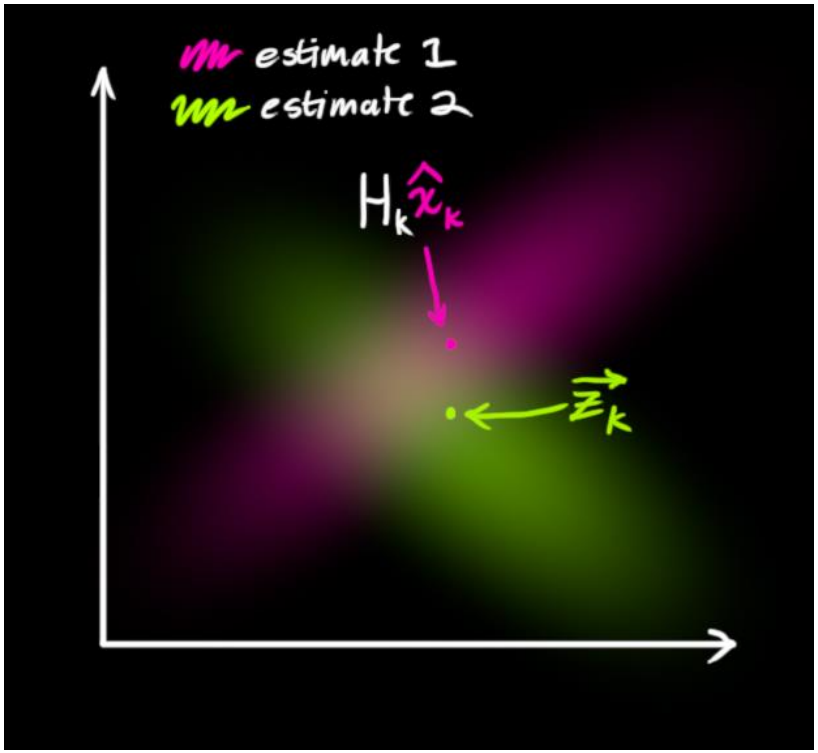
Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

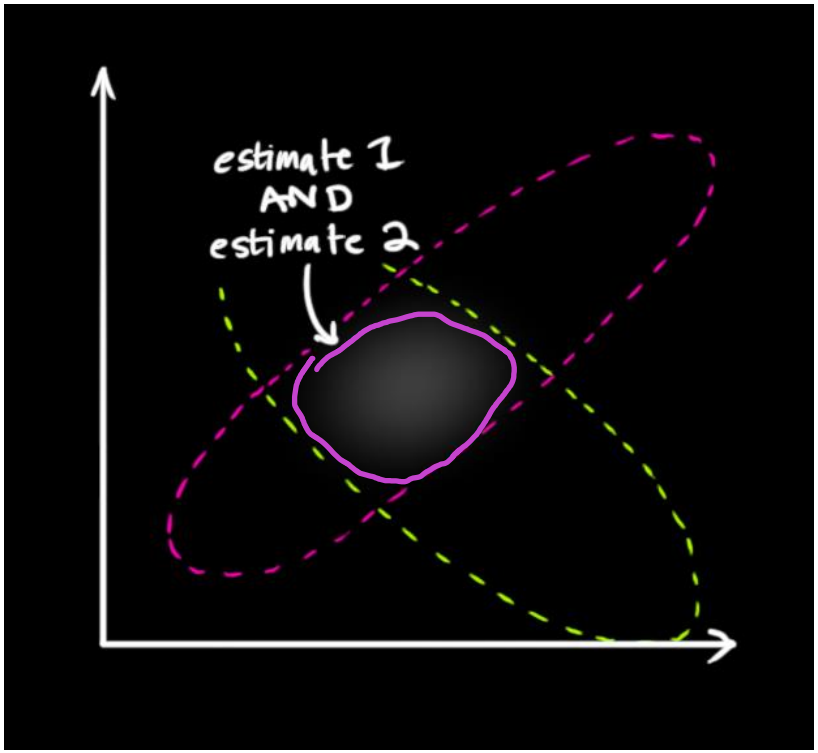


State Estimation Methods

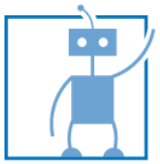
Kalman Filter

- Two Gaussians: One surrounding the mean of our transformed prediction, and one surrounding the actual sensor reading we got.





Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- Two Gaussians: One surrounding the mean of our transformed prediction, and one surrounding the actual sensor reading we got. Combining both:

$$\mathcal{N}(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

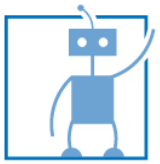
$$\mathcal{N}(x, \mu_0, \sigma_0) \cdot \mathcal{N}(x, \mu_1, \sigma_1) \stackrel{?}{=} \mathcal{N}(x, \mu', \sigma')$$

$$\mathbf{k} = \frac{\sigma_0^2}{\sigma_0^2 + \sigma_1^2}$$

$$\mu' = \mu_0 + \mathbf{k}(\mu_1 - \mu_0)$$

$$\sigma'^2 = \sigma_0^2 - \mathbf{k}\sigma_0^2$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

kalman gain

- Two Gaussians: One surrounding the mean of our transformed prediction, and one surrounding the actual sensor reading we got. Combining both. Rewrite into matrix form.

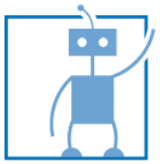
$$\mathbf{K} = \Sigma_0(\Sigma_0 + \Sigma_1)^{-1}$$

$$\vec{\mu}' = \vec{\mu_0} + \mathbf{K}(\vec{\mu_1} - \vec{\mu_0})$$

$$\Sigma' = \Sigma_0 - \mathbf{K}\Sigma_0$$

$$K_n = \frac{Uncertainty\ \ in\ \ Estimate}{Uncertainty\ \ in\ \ Estimate\ +\ Uncertainty\ \ in\ \ Measurement}$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- Two Gaussians: One surrounding the mean of our transformed prediction, and one surrounding the actual sensor reading we got. Combining both. Rewrite into matrix form. Putting everything together to develop the update step equations:

$$\mathbf{K} = \Sigma_0(\Sigma_0 + \Sigma_1)^{-1}$$

$$\vec{\mu}' = \vec{\mu_0} + \mathbf{K}(\vec{\mu_1} - \vec{\mu_0})$$

$$\Sigma' = \Sigma_0 - \mathbf{K}\Sigma_0$$

$$(\mu_0, \Sigma_0) = (\mathbf{H}_k\hat{\mathbf{x}}_k, \mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T)$$

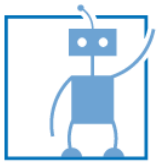
$$(\mu_1, \Sigma_1) = (\vec{\mathbf{z}_k}, \mathbf{R}_k)$$

$$\mathbf{H}_k\hat{\mathbf{x}}'_k = \mathbf{H}_k\hat{\mathbf{x}}_k + \mathbf{K}(\vec{\mathbf{z}_k} - \mathbf{H}_k\hat{\mathbf{x}}_k)$$

$$\mathbf{H}_k\mathbf{P}'_k\mathbf{H}_k^T = \mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T - \mathbf{K}\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T$$

$$\mathbf{K} = \mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

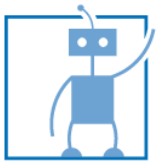
- Two Gaussians: One surrounding the mean of our transformed prediction, and one surrounding the actual sensor reading we got. Combining both. Rewrite into matrix form. Putting everything together to develop the update step equations:

$$\hat{\mathbf{x}}'_k = \hat{\mathbf{x}}_k + \mathbf{K}'(\vec{\mathbf{z}_k} - \mathbf{H}_k\hat{\mathbf{x}}_k)$$

$$\mathbf{P}'_k = \mathbf{P}_k - \mathbf{K}'\mathbf{H}_k\mathbf{P}_k$$

$$\mathbf{K}' = \mathbf{P}_k\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter

- Perform Prediction Step and Update step iteratively. Start with priors.

Prediction:

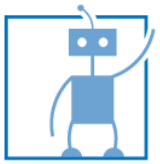
$$\hat{\mathbf{x}}_k = \mathbf{F}_k\hat{\mathbf{x}}_{k-1} + \mathbf{B}_k\vec{\mathbf{u}_k}$$
$$\mathbf{P}_k = \mathbf{F_k}\mathbf{P}_{k-1}\mathbf{F}_k^T + \mathbf{Q}_k$$

Update:

$$\hat{\mathbf{x}}'_k = \hat{\mathbf{x}}_k + \mathbf{K}'(\vec{\mathbf{z}_k} - \mathbf{H}_k\hat{\mathbf{x}}_k)$$
$$\mathbf{P}'_k = \mathbf{P}_k - \mathbf{K}'\mathbf{H}_k\mathbf{P}_k$$

$$\mathbf{K}' = \mathbf{P}_k\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

Images: Bzarg, phi{losophy,sics,ction}, https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/



State Estimation Methods

Kalman Filter – Code comparison

```
%% 1D-Kalman Filter
% Aufbau und Erläuterung nach Udacity Kurs CS373 taught by Sebastian Thrun
% Paul Balzer
clear all, clc

%% Ausgangsbedingungen
% sigma=Standardabweichung, m = Mittelwert
sigma_mess = 4; % Standardabweichung Messung
sigma_move = 2; % Standardabweichung Bewegung

mu = 0; % Startposition
sig = 10000; % Unsicherheit zu Beginn

%% Kalman-Berechnung

messung = [5, 6, 7, 9, 10];
bewegung= [1, 1, 2, 1, 1];

for i=1:length(messung)
 [mu,sig]=predict(mu,sig,bewegung(i),sigma_move);
 disp(['Predict: ' num2str(mu) ', ' num2str(sig)])
 [mu,sig]=update(mu,sig,messung(i),sigma_mess);
 disp(['Update: ' num2str(mu) ', ' num2str(sig)])
end
```

```
function [new_mean, new_var]=predict(mean1,var1,mean2,var2)
%% [new_mean, new_var]=predict(mean1,var1,mean2,var2)
% Berechnet neuen Mittelwert und neue Varianz nach Bewegung mit Mittelwert
% und Varianz von Ausgangsposition und Unsicherheit der Bewegung sowie
% vorhergesagter Bewegungsentfernung
% new_mean = mean1 + mean2;
% new_var = var1 + var2;
new_mean=mean1+mean2;
new_var =var1+var2;
```

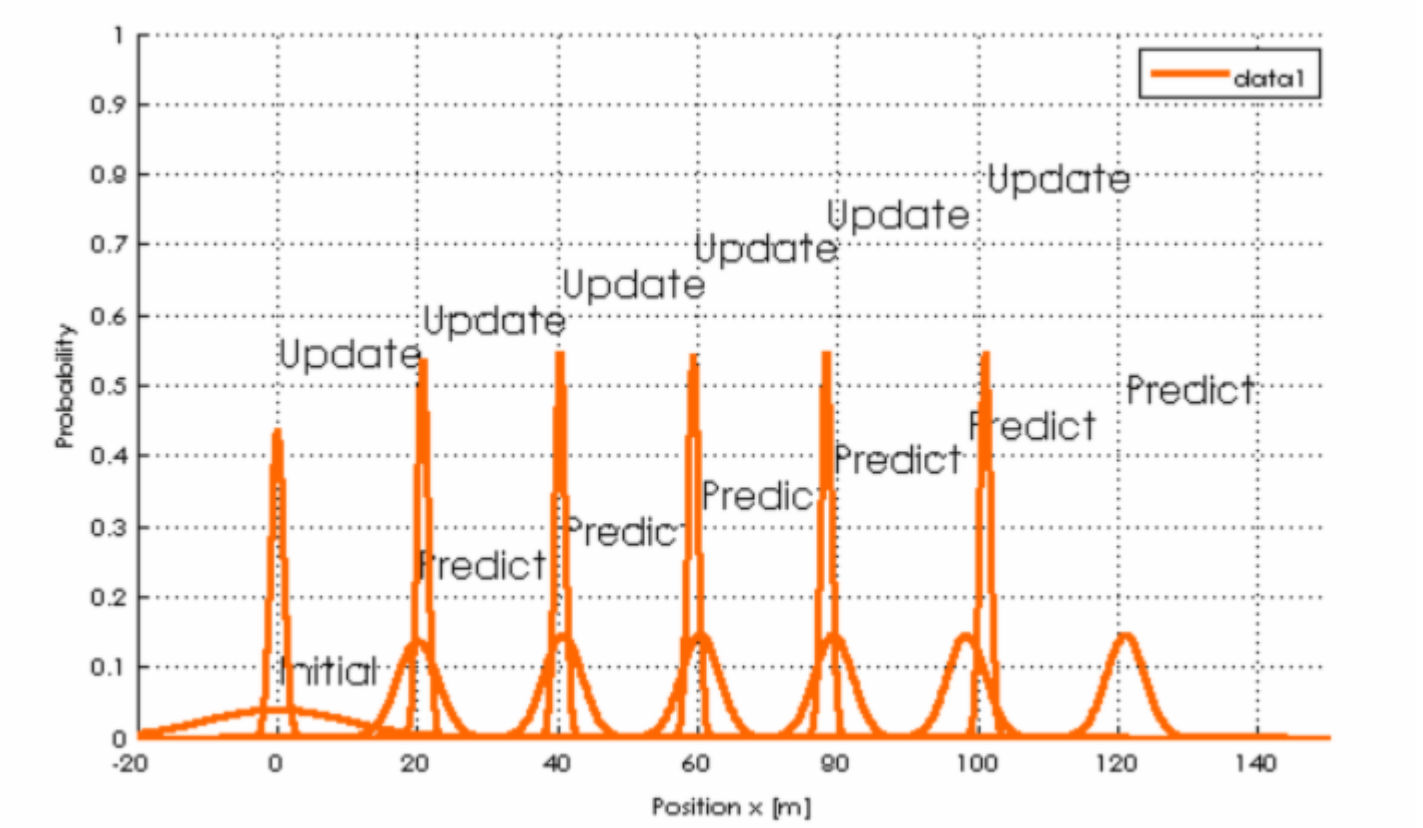
```
function [new_mean, new_var]=update(mean1,var1,mean2,var2)
%% [new_mean, new_var]=update(mean1,var1,mean2,var2)
% Berechnet geschätzte Position nach Messung der Position

new_mean=(var2*mean1 + var1*mean2) / (var1+var2);
new_var = 1/(1/var1 +1/var2);
```

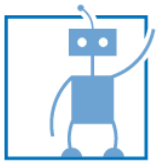
Code from https://www.cbcity.de/das-kalman-filter-einfach-erklaert-teil-1

State Estimation Methods

Kalman Filter – Impact update vs prediction



Code from https://www.cbcity.de/das-kalman-filter-einfach-erklaert-teil-1



Gereon Hinz

STTech GmbH

Floriansmühlstraße 8 - 80939 München

Tel: 089/905499430

gereon.hinz@sttech.de

www.sttech.de