Incremental recognition of multi-object behaviour using hierarchical probabilistic models

Frank-Michael ZIMMER and Bernd NEUMANN
Department of Informatics, University of Hamburg

Abstract. In this contribution we present a new methodological framework and first results for real-time monitoring of object behaviour in aircraft servicing scenes, such as arrival preparation, unloading, tanking and others, based on video streams from several cameras. The focus is on incremental real-time interpretation of multiple object tracks. We show that the temporal structure of complex, partially coordinated object behaviour such as aircraft servicing can be modelled by a Bayesian Compositional Hierarchy (BCH). This is a recently developed kind of Bayesian Network where aggregates are modelled with unrestricted distributions, whereas the dependency structure between aggregates is restricted to correspond to the tree structure of the compositional hierarchy. This allows efficient updating when evidence is incorporated incrementally. For the domain of service operations, a BCH has been constructed for modelling the durations of activities and delays between them. The BCH is primarily used to provide a ranking of alternative partial interpretations and control the interpretation process according to the beam search paradigm. In addition, a BCH can provide estimates of missing data based on current evidence, for example, regarding the duration of a servicing operation. We explain the structure of aggregates constituting the aircraft servicing BCH and demonstrate evidence-based updates as well as predictions.

Keywords. Scene interpretation, multi-object behaviour, probabilistic models

Introduction

Aircraft servicing is an example of multi-object behaviour with interesting challenges for modelling and recognition. A single turnaround consists of a large number of activities, beginning with arrival preparation on the apron, diverse service activities such as unloading, loading, tanking and catering, and ending with a push-back. Most activities can be decomposed into several subactivities and sub-subactivities, hence it is natural to think of a turnaround in terms of a hierarchy of activities. Many of the components of the hierarchy are loosely related, for example tanking and catering are unrelated except for being part of a turnaround. Other components are strongly related, for example the arrival of a tanker, the tanking process and the departure of the tanker follow a strict sequence. Many details are probabilistic in nature, in particular the durations of and the delays between activities. The positions taken by servicing vehicles relative to an aircraft are essentially predetermined, but the paths leading to these positions may vary considerably. In addition to objects participating in regular servicing operations, there may be other

"spurious" objects in the servicing area for unknown purposes, e.g. a technician performing a check.

The goal in this work[1] is to monitor servicing operations as a support for airport logistics. To this end, it is necessary to recognise individual operations and provide real-time estimates about the future of a turnaround. We focus here on the high-level part of activity recognition which takes the results of object tracking and recognition as input and generates high-level interpretations in terms of instantiated activity models. The interpretation process poses several challenges which in our belief are typical for real-time understanding of multi-object behaviour. First, primitive data entering the interpretation process are often ambiguous regarding their role in a hierarchical activity model. For example, a vehicle stopping on the apron may mark the beginning of several possible servicing activities. It highly depends on the available real-time context, to which degree this inherent ambiguity can be narrowed down. Second, real-time processing does not allow a wait-and-see strategy where data is collected until enough evidence for safe decisions is available. Instead, the interpretation process must be able to entertain competing partial interpretations, each providing its own context for guiding the interpretation of the next incremental evidence.

In principle, Bayesian Networks provide a well-understood way for obtaining estimates from incremental evidence. Here we are faced, however, with an application domain where activities are naturally modelled as a multilevel hierarchical structure composed of structured entities, called aggregates. Compositional hierarchies have been employed for high-level scene interpretation by many researchers [1, 2, 3, 4, 5] with basically non-probabilistic (crisp) frame-based representations, as commonly used in AI. Rimey [6] was the first to model compositional hierarchies with tree-shaped Bayesian Networks (BNs), requiring parts of an aggregate to be conditionally independent. Koller and coworkers [7, 8] extended BNs in an object-oriented manner for the representation of structured objects. Their Probabilistic Relational Models allow to augment a crisp relational structure with an arbitrary probabilistic dependency structure. Gyftodimos and Flach [9] introduce hierarchical BNs for multiple levels of granularity. While these contributions improve the expressive power of BNs, they do not specifically support compositional hierarchies of aggregates as required for context modelling in scene interpretation. For this purpose, Bayesian Compositional Hierarchies (BCHs) have been developed and first applied to static scenes [10]. An interesting alternative approach has been published by Mumford and Zhu [11] where a grammatical formalism takes the place of hierarchical knowledge representation and parsing algorithms are applied for scene interpretation, leading to efficient processing, but complicating the integration with large-scale knowledge representation.

In this contribution, the application of a BCH to real-time scene interpretation is explored for the first time. Aggregates are used to represent the temporal structure of activities and their constituents in an object-centered manner. At the lowest level, the parts of an aggregate correspond to primitive events, such as `Tanker-Stopped-Inside-Tanking-Zone` or `Stop-Beacon`, provided by the tracking system and the middle layer of the system. In general, a primitive event can be part of several aggregates, including a "clutter" model, and the probabilistic model is used to rank alternative evidence assignments. High-ranking alternatives are maintained in a beam search and provide MAP estimates of alternative scene interpretations throughout the

[1] This work was partially supported by the EC, Grant 214975, Project Co-Friend.

process. Besides incoming evidence, the progressing real-time can also be exploited for updating a BCH to the effect that missing evidence can only be expected in the future.

In Section 1, we first describe the probabilistic framework for incremental scene interpretation using beam search in a general form. In Section 2, we present the BCH as a tree-shaped Bayesian Network with aggregates of arbitrary complexity as nodes. We also show that Gaussian probability density functions (PDFs) give rise to very efficient update operations. In Section 3, we present examples from the aircraft service domain and demonstrate the predictive power of the model as well as the ranking of alternative partial interpretations. We conclude with a summary and an outlook on future work.

1. Probabilistic framework

1.1. Probabilistic scene model

In a general form, probabilistic scene interpretation can be modelled as evidence-based reasoning with large joint probability distributions (JPDs). Let us assume that the task is to determine which of M alternative models applies to a scene. Then a generative probabilistic model for a scene can be written as

$$P_{scene} = P(S)\ P^{(m)}(\underline{X}_1^{(m)} \ldots \underline{X}_{N^{(m)}}^{(m)}\ \underline{Y}_1^{(m)} \ldots \underline{Y}_{K^{(m)}}^{(m)})\ P_{clutter} \quad (1)$$

The random variable S with values from 1..M selects a model m with prior probability $P(S=m) = q_m$. Each model is described by a JPD consisting of hidden variables $\underline{X} = [\underline{X}_1 .. \underline{X}_N]$ and observable variables $\underline{Y} = [\underline{Y}_1 .. \underline{Y}_K]$. The indices suggest distinct conceptual objects, each described by a vector of random variables (indicated by the underline). Values for observable variables are provided by evidence from low-level processing, values of hidden variables are determined by probabilistic inference. $P_{clutter}$ is a catch-all distribution for evidence not fitting a model. In our temporal model for the aircraft servicing domain, the observables could correspond to time points marking a primitive event such as `Airplane-Stopped-Inside-ERA` (ERA is the entrance-restricted area around an aircraft), whereas hidden variables could describe beginning and duration of higher-level activities such as `Arrival-Preparation`. $P_{clutter}$ could simply be a JPD modelling the occurrence of "unexplainable" evidence objects during a turnaround as independent events.

To guide the interpretation process, we are interested in a ranking of alternative interpretations for given partial evidence $\underline{e}$. Alternatives do not only arise from the models 1 .. M but also from alternative assignments of evidence within a model. For example, a `Vehicle-Enters-ERA` event can be part of several service activities of a turnaround, in particular, if the type of vehicle is uncertain. Also, since low-level processing is not perfect, and tracking errors as well as misclassifications occur. To simplify the notation, we enumerate alternative evidence assignments together with alternative models using the index n. Further alternatives arise from assigning some of the evidence - say $\underline{e}_n^+$ - to the model and the rest - say $\underline{e}_n^-$ - to clutter, possibly different for each model. Hence the ranking R_n of a scene model n is given by the probability of that model of having generated $\underline{e}_n^+$ as part of the service model and $\underline{e}_n^-$ as clutter. This is captured by the following equation:

$$R_n = q_n \, P^{(n)}(\underline{e}_n^+) \, P_{clutter}(\underline{e}_n^-) \tag{2}$$

Eq. (2) shows that alternative rankings can be determined from Eq. (1) by marginalising the observables of each model m which have been chosen for evidence assignment, and computing the resulting probabilities.

The final interpretation is given in terms of values $\underline{x}_n$* for hidden variables and evidence assignment $\underline{e}_n$ for observables of the highest-ranking model obtained by the following maximizations:

$$n^* = \arg\max_n (q_n P^{(n)}(\underline{e}_n^+) P_{clutter}(\underline{e}_n^-)) \tag{3}$$

$$[\underline{X} = \underline{x}^*, \underline{Y} = \underline{e}^+] = \arg\max_{\underline{x}} (q_{n^*} P^{(n^*)}(\underline{x}, \underline{e}_{n^*}^+)) \tag{4}$$

Note that the probabilistic model given by Eq. (1) does not explicitely account for missing evidence, for example due to occlusion or tracking limitations. To deal with this, the range of observables could be extended to include "missing evidence" as a possible "value", but an assignment and probabilistic appreciation will necessarily depend on the context. The issue of missing information will not be treated in the sequel.

1.2. Real-time updates

In our application, a scene model as given by Eq. (1) will involve temporal random variables representing observable events on a quantitative time scale relative to some common reference event, for example relative to an initial observation. Real-time processing using such a model implies that we have a current time t_c which progresses as we observe a concrete scene, and that modelled events not observed so far are bound to happen at times $t > t_c$, if at all. This should influence our ranking of alternatives to the effect that reduced chances for an event cause a reduced ranking.

Let $\underline{e}$ be evidence assigned up to time t_c, and $\underline{T}_n \subseteq \underline{Y}$ be unassigned temporal observables of a service model. Then the rank of model n at time t_c is given by

$$R_n(t_c) = q_n P^{(n)}(\underline{e}_n^+, \underline{T}_n > t_c) P_{clutter}(\underline{e}_n^-) \tag{5}$$

Eq. (5) shows that the ranking of an alternative model changes according to its share in the probability space for the remaining temporal variables. This refines Eq. (2) which implied that the complete probability space was left for unassigned variables. Note that real-time updating does not apply to hidden temporal variables which may take values $t < t_c$.

1.3. Interpretation using beam search

In the preceding sections, we have shown how real-time incremental evidence assignments in a probabilistic framework provide a dynamic ranking for alternative scene models. This can be exploited by a parallel search strategy called beam search

[15], where only promising alternative partial interpretations are kept (in the "beam"), while improbable ones are discarded.

For real-time scene interpretation with beam search, the following steps have to be executed:

A Initialise the beam with all alternative models given by Eq. (1). Wait for an initial event to start the real-time clock.
B Wait for next evidence $\underline{e}$. While waiting, perform real-time updates according to Eq. 5 following an update schedule.
C Determine possible assignments of $\underline{e}$ for each model in the beam, clone models in case of multiple assignments.
D Rank models using Eq. 5, discard unlikely models from the beam.
E Repeat B to E until all evidence is assigned.
F Select highest-ranking model and determine final interpretation using Eq. 3.

Efficient storage of P_{scene} and computation of the marginalisations in Eqs. 2 and 5 may easily become a bottleneck for realistic tasks. Therefore, Bayesian Network technology is required and the dependency structure of object properties plays an important part. In the following section, we will present Bayesian Compositional Hierarchies [10] which allow arbitrary dependencies within aggregates but are restricted to a tree-shaped dependency structure between aggregates, thus providing efficient computational procedures in tune with compositional hierarchies.

2. Bayesian Compositional Hierarchies

As pointed out in the introduction, compositional hierarchies are often used as a natural conceptual framework for scene interpretation tasks. It is therefore useful to adapt the general approach decribed in Section 1 to hierarchical models. Rimey [6] has been the first in Computer Vision to develop tree-shaped Bayesian Networks (BNs) for compositional hierarchies. To ensure efficient processing, he had to assume that parts of an aggregate are statistically independent given the parent aggregate. In [10] a more powerful hierarchical probabilistic model has been presented, called Bayesian Compositional Hierarchy (BCH). In the following, we briefly summarise the definition of a BCH for arbitrary probability distributions. Thereafter, we describe the structure of a Gaussian BCH which is the kind used for modelling the temporal structure of aircraft services in our work.

2.1. General structure of a BCH

A BCH is a probabilistic model of a compositional hierarchy. It consists of aggregates, each modelled individually by an unrestricted JPD in an object-centered manner. The hierarchy is formed by using the aggregate headers as part descriptions in aggregates of the next hierarchical level, abstracting from details of parts at the lower level.

Figure 1 illustrates the schematic structure of a BCH. Each aggregate is described by a JPD $P(\underline{A}\ \underline{B_1}..\underline{B_K}\ \underline{C})$ where $\underline{A}$ is the aggregate header providing an external description to the next higher level, $\underline{B_1}..\underline{B_K}$ are descriptions of the parts, and $\underline{C}$ expresses conditions on the parts. The hierarchy is constructed by taking the aggregate

headers at a lower level as part descriptions at the next higher level, hence $\underline{B}_1^{(1)} = \underline{A}^{(2)}$ etc.

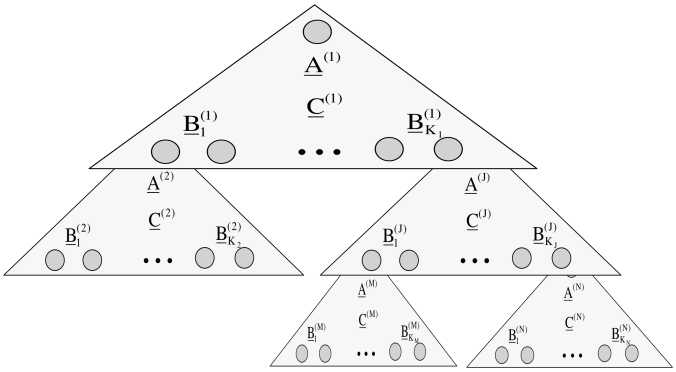

Figure 1. Structure of a BCH. Triangles represent aggregates, circles represent parts. Aggregate models overlap, their headers represent parts of aggregates at the next higher level.

In our aircraft servicing domain, for example, a `Turnaround` aggregate consists of a header which provides an external description of a turnaround in terms of its duration (abstracting from details about the parts), and an internal description of the temporal structure of the three parts `Arrival`, `Services` and `Departure`. The parts are also described as aggregates themselves, for example `Arrival` is an aggregate with parts `Arrival-Preparation`, `Airplane-Enters-ERA` and `Stop-Beacon`. The complete hierarchy is shown in Table 1.

In general, the JPD of a complete hierarchy is given by

$$P(\underline{A}^{(1)} .. \underline{A}^{(N)}) = P(\underline{A}^{(1)}) \prod_{i=1..N} P(\underline{B}_1^{(i)} .. \underline{B}_{K_i}^{(i)} \underline{C}^{(i)} \mid \underline{A}^{(i)}) \tag{6}$$

This remarkable formula shows that the JPD of a BCH can be easily constructed from individual aggregate representations, and belief updates can be performed by propagation along the tree structure. Let $P'(B_i)$ be an update of $P(B_i)$, by evidence or propagation from its parts below. Then the updated aggregate JPD is

$$P'(\underline{A}\ \underline{B}_1 .. \underline{B}_K\ \underline{C}) = P(\underline{A}\ \underline{B}_1 .. \underline{B}_K\ \underline{C})\ P'(\underline{B}_i)/P(\underline{B}_i) \tag{7}$$

A similar equation holds when P(A) is updated by propagation from its parent above.

Storage and updating operations for large hierarchies can be computationally very expensive. We have therefore developed an implementation for aggregates with multivariate Gaussian distributions. The propagation formulas for a Gaussian BCH are summarised in the following.

2.2. Propagation in a Gaussian BCH

Roughly symmetric, unimodal distributions can often be approximated by a Gaussian in a range corresponding to $-2\sigma .. +2\sigma$, where σ is the standard deviation.

Table 1. Aggregates of `Turnaround` hierarchy

```
Turnaround
        Arrival
                Arrival-Preparation
                        GPU-Enters-GPU-Zone
                        GPU-Stopped-Inside-GPU-Zone
                        Drop-Chocks
                Airplane-Enters-ERA
                Airplane-Stopped-Inside-ERA
                Stop-Beacon
        Services
                Passenger-Activity
                        Passenger-Stairs-Enters-PS-Zone
                        Passenger-Stairs-Stopped
                Unload-Right
                        Unload-Right-AFT
                                Loader-Enters-Right-AFT-LD-Zone
                                Transporter-Enters-Right-AFT-TS-Zone
                                Unload-Motion-Right-AFT-Belt
                                Transporter-Leaves-Right-AFT-TS-Zone
                                Loader-Leaves-Right-AFT-LD-Zone
                        Unload-Right-FWD
                                Loader-Enters-Right-FWD-LD-Zone
                                Transporter-Enters-Right-FWD-TS-Zone
                                Unload-Motion-Right-FWD-Belt
                                Transporter-Leaves-Right-FWD-TS-Zone
                                Loader-Leaves-Right-FWD-LD-Zone
                Air-Conditioning
                        Air-Conditioning-Unit-Enters-Air-Conditioning-Zone
                        Air-Conditioning-Unit-Stopped
                        Air-Conditioning-Unit-Plugged-In
                Catering
                        Catering-Van-Enters-Catering-Zone
                        Catering-Van-Stopped
                        Catering-Van-Raised
                Refuelling
                        Tanker-Enters-Tanking-Zone
                        Tanker-Stopped
                        Pumping-Operation
                Replace-Drinking-Water
                        Drinking-Water-Tank-Enters-Drinking-Water-Zone
                        Drinking-Water-Tank-Stopped
                        Drinking-Water-Plugged-In
                Waste-Removal
                        Waste-Removal-Vehicle-Enters-Waste-Removal-Zone
                        Waste-Removal-Vehicle-Stopped
                        Waste-Removal-Unit-Plugged-In
                Load-Right
                        Load-Right-AFT
                                Loader-Enters-Right-AFT-LD-Zone
                                Transporter-Enters-Right-AFT-TS-Zone
                                Load-Motion-Right-AFT-Belt
                                Transporter-Leaves-Right-AFT-TS-Zone
                                Loader-Leaves-Right-AFT-LD-Zone
                        Load-Right-FWD
                                Loader-Enters-Right-FWD-LD-Zone
                                Transporter-Enters-Right-FWD-TS-Zone
                                Load-Motion-Right-FWD-Belt
                                Transporter-Leaves-Right-FWD-TS-Zone
                                Loader-Leaves-Right-FWD-LD-Zone
        Departure
                Start-Beacon
                Pushback
```

Multivariate Gaussian aggregate models can be compactly represented by means and covariance matrices, and propagation in a BCH can be performed very efficiently by closed-form solutions, as shown in the following.

Let G = [E F] be a vector of Gaussian random variables representing an aggregate. Let F be the subset whose distribution is changed by evidence or incoming propagation. F can be the aggregate header in the case of downward propagation or a part header in the case of upward propagation. We want to compute the effect of the changed distribution of F on G. Before propagation, the distribution of G is $P(\underline{G}) = N(\underline{\mu}_G, \Sigma_G)$ where $\underline{\mu}_G$ is the mean vector and Σ_G the covariance matrix. The partitions corresponding to E and F, respectively, are denoted as shown:

$$\Sigma_G = \begin{bmatrix} \Sigma_E & \Sigma_{EF} \\ \Sigma_{EF}^T & \Sigma_F \end{bmatrix} \qquad \underline{\mu}_G = \begin{bmatrix} \underline{\mu}_E \\ \underline{\mu}_F \end{bmatrix}$$

For a probability update, we assume that the distribution of F is changed to $P'(\underline{F}) = N(\underline{\mu}_F', \Sigma_F')$. Then the new distribution of G is $P'(\underline{G}) = N(\underline{\mu}_G', \Sigma_G')$ with

$$\Sigma_G' = \begin{bmatrix} \Sigma_E' & \Sigma_{EF}' \\ \Sigma_{EF}'^T & \Sigma_F' \end{bmatrix} \qquad \underline{\mu}_G' = \begin{bmatrix} \underline{\mu}_E' \\ \underline{\mu}_F' \end{bmatrix} \quad \text{where}$$

$$\Sigma_E' = \Sigma_E - \Sigma_{EF} \Sigma_F^{-1} \Sigma_{EF}^T + \Sigma_{EF} \Sigma_F^{-1} \Sigma_F' \Sigma_F^{-1} \Sigma_{EF}^T \tag{8}$$

$$\Sigma_{EF}' = \Sigma_{EF} \Sigma_F^{-1} \Sigma_F' \tag{9}$$

$$\underline{\mu}_E' = \underline{\mu}_E + \Sigma_{EF} \Sigma_F^{-1} (\underline{\mu}_F' - \underline{\mu}_F) \tag{10}$$

The Gaussian updating rules in Eqs. (8) to (10) have been first presented in [10]. It is evident that both upward and downward propagation for an aggregate with random variables A B_1 ... B_K C can be performed by fairly simple matrix computations.

Multivariate Gaussians are also very convenient for implementing the interpretation procedure by beam search as described in Section 1. The marginalisations required for ranking alternative interpretations according to Eq. (2) are directly available from the aggregate covariances, and the final maximising interpretation according to Eq. (3) can be given in terms of the mean values of hidden variables.

There are, however, clear limitations of the applicability of multivariate Gaussian BCHs, for example in connection with discrete random variables, range-limited flat distributions or the truncated distributions arising in real-time updates according to Eq. (5). In some cases it may be possible though to use Gaussians as approximations. This will be shown in the next section where the temporal structure of aircraft services is modelled by a BCH based on a multivariate Gaussian distribution.

3. Temporal models for aircraft servicing

To perform real-time interpretation of aircraft servicing operations, a BCH has been designed consisting of the aggregates shown in Table 1. The leaves of the hierarchy are primitive aggregates without parts which will be instantiated by evidence from lower-level processing. Spatial information is expressed in terms of qualitative positions in predefined zones. For example `Airplane-Enters-ERA` specifies that

an airplane enters the entrance-restricted area (ERA) marked on the apron for aircraft servicing. Other zones, e.g. the loading zone in `Loader-Enters-Right-AFT-LD-Zone`, are defined relative to the aircraft and depend on its type.

Each (non-primitive) aggregate specifies the temporal structure of its parts in terms of correlated random variables for durations and delays. Figure 2 illustrates the structure of the aggregate `Arrival` as an example. The aggregate header is a random variable for the duration of `Arrival`. Its value is defined as the sum of the duration of `Arrival-Preparation` and the delays of the point events `Airplane-Enters-ERA`, `Airplane-Stopped-Inside-ERA` and `Stop-Beacon`.

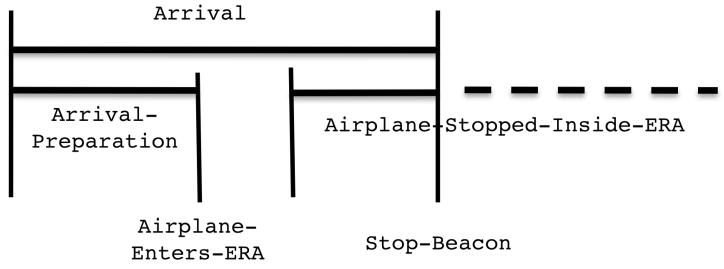

Figure 2. Temporal structure of the aggregate `Arrival`.

Figure 3 shows means and covariance of the Gaussian JPD, estimated from the data of 52 turnarounds. Note that the beginning and ending of `Arrival` is determined by the random variables describing the parts, leading to a singular covariance - this does not jeopardise the updating procedure. The positive correlation between some of the durations and delays reflects the observation that activities in some turnarounds are generally faster than in others.

	mean	covariance				
Arrival	17	32	11	18	1,5	1,5
Arrival-Preparation	6	11	9	2	0	0
Airplane-Enters-ERA-Delay	2	18	2	16	0	0
Airplane-Stopped-Inside-ERA-Delay	1	1,5	0	0	1	0,5
Stop-Beacon-Delay	1	1,5	0	0	0,5	1

Figure 3. Multivariate Gaussian JPD for `Arrival` as a sum of the duration of `Arrival-Preparation` and the delays of the point events `Airplane-Enters-ERA` and `Stop-Beacon` (units in minutes).

All aggregate models have a similar structure, with activities described by their durations and related to each other by delays. Gaussians are used with the understanding that only the range $-2\sigma \ldots +2\sigma$ is valid in the model. To ensure that durations of activities take only positive values, their models are constrained by $\mu > 2\sigma$.

To demonstrate the predictive power of the BCH, the estimated timeline for turnaround events, and the remaining uncertainty (measured in standard deviations) has been determined for two cases, (i) after observing the very first event, `GPU-Enters-GPU-Zone`, and (ii) after observing all events up to a late `Aircraft-Enters-ERA`, see Table 2. Note that extended activities are marked with the suffix -Beg and -End indicating begin and end, respectively, while point events are marked with the suffix -Eve. It can be seen that observations in Case 2 significantly change the expectations of

future events due to the correlations within aggregate models. Also, as expected, the uncertainty of estimates decreases with additional evidence.

Table 2. Estimated timeline of a turnaround after initial observation (Case 1) and after observations up to `Airplane-Enters-ERA` (Case 2). Columns show times T and uncertainties of estimates D (in standard deviations).

	Case 1		Case 2	
	T	D	T	D
Turnaround-Beg	**0**	**0**	**0**	**0**
Arrival-Beg	0	0	**0**	**0**
Arrival-Preparation-Beg	0	0	**0**	**0**
GPU-Enters-GPU-Zone-Eve	0	0	**0**	**0**
GPU-Stopped-Inside-GPU-Zone-Beg	1	0,5	**1**	**0**
Drop-Chocks-Eve	6	3	**6**	**0**
Arrival-Preparation-End	6	3	**6**	**0**
Airplane-Enters-ERA-Eve	9	6	**15**	**0**
Airplane-Stopped-Inside-ERA-Beg	9	6	15	0
Stop-Beacon-Eve	17	6	23	1
Arrival-End	17	6	23	2
Services-Beg	19	8	26	3
Passenger-Activity-Beg	19	8	26	3
Passenger-Stairs-stopped-Inside-PS-Zone-Beg	19	8	26	3
Passenger-Stairs-stopped-Inside-PS-Zone-End	55	16	62	15
Passenger-Activity-End	58	17	65	15
Unload-Right-Beg	23	9	30	5
Unload-Right-AFT-Beg	23	9	30	5
Loader-Stopped-Inside-Right-AFT-LD-Zone-Beg	23	9	30	5
Transp.-Stopped-Inside-Right-AFT-TS-Zone-Eve	25	9	32	5
Unload-Motion-Right-AFT-Belt-Beg	29	9	36	6
Unload-Motion-Right-AFT-Belt-End	39	10	46	7
Transp.-Stopped-Inside-Right-AFT-TS-Zone-End	41	10	48	7
Loader-Stopped-Inside-Right-AFT-LD-Zone-End	43	10	50	7
Unload-Right-AFT-End	43	10	50	7
Unload-Right-FWD-Beg	24	12	31	10
Loader-Stopped-Inside-Right-FWD-LD-Zone-Beg	24	12	31	10
Transp.-Stopped-Inside-Right-FWD-TS-Zone-Beg	25	12	33	10
Unload-Motion-Right-FWD-Belt-Beg	29	13	36	11
Unload-Motion-Right-FWD-Belt-End	39	13	46	11
Transp.-Stopped-Inside-Right-FWD-TS-Zone-End	41	13	48	11
Loader-Stopped-Inside-Right-FWD-LD-Zone-End	43	14	50	12
Unload-Right-End	43	13	50	11
Refuelling-Beg	33	31	40	30
Tanker-Stopped-Inside-Tanking-Zone-Beg	33	31	40	30
Pumping-Operation-Beg	36	31	43	30
Pumping-Operation-End	43	31	50	30
Tanker-Stopped-Inside-Tanking-Zone-End	47	31	54	30
(further service operations omitted for brevity)				
Services-End	55	23	62	22
Departure-Beg	57	24	69	21
Start-Beacon	57	24	69	21
Pushback-Beg	58	25	70	22
Pushback-End	60	25	72	22
Departure-End	60	25	72	22
Turnaround-End	60	25	72	22

We now describe a concrete scene interpretation task based on real data to demonstrate the ranking provided by the BCH in a beam search. The input data has been obtained from one of 80 turnarounds recorded at the Blagnac Airport in Toulouse by low-level processing of project partners in France and England. Interpretation with beam search was performed by the system SCENIOR developed in the group of the authors [13].

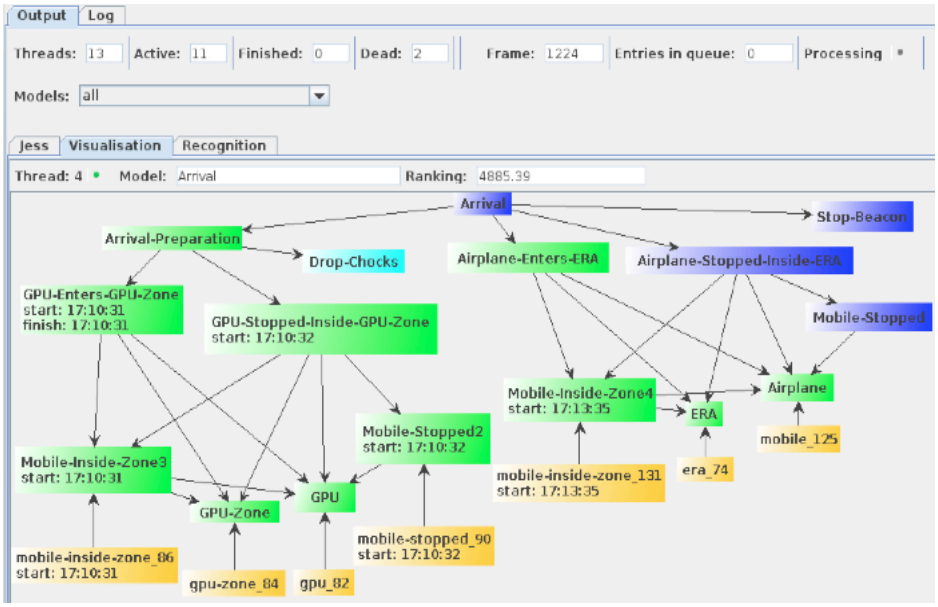

Figure 4. Interpretation alternative No. 4 generated by SCENIOR after 3 minutes real-time

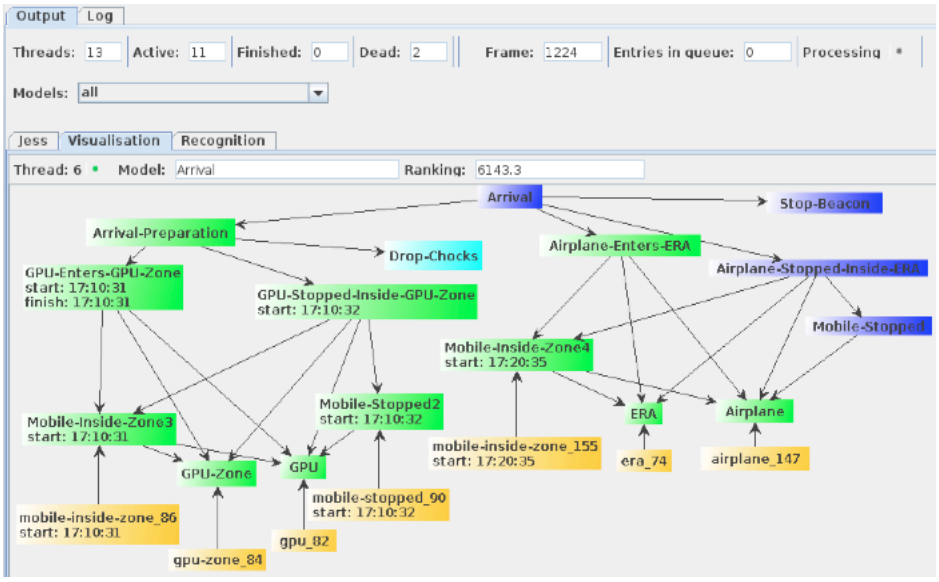

Figure 5. Interpretation alternative No. 6 generated by SCENIOR after 10 minutes

Here, we describe the initial phase where two competing interpretations of the activity `Arrival` are presented from 13 alternative interpretations generated by SCENIOR. Figures 4 and 5 show the evidence received so far (yellow boxes at the bottom), the instantiated parts of the arrival model (green boxes) and expected further events (dark blue boxes). The `Drop-Chocks` event could not be observed and was inferred from the context. The figures do not show any of the several clutter events which did not fit the partially instantiated models.

The clutter probability has been set to 0,01 to favour more complete interpretations. Since the ratings are naturally decreasing with each step and may reach very small numbers, a scaling factor of 100 is applied at each step. Thus, a clutter assignment renders ratings nominally unchanged.

The main difference between the interpretations is an erroneous `Airplane-Enters-ERA` event generated by low-level processing for a tanker crossing the ERA shortly before the arrival of the airplane. Figure 6 shows the corresponding video frames taken by one of the eight cameras. The crossing tanker is visible in the far background of the image on the left.

Figure 6. Snapshots of the ERA (Entrance Restricted Area) after completing `Arrival-Preparation`. The GPU (Ground Power Unit) and shocks are in place. The tanker crossing the ERA in the background (left) causes an erroneous interpretation thread (see text).

Table 3. Initial ratings of the two alternative interpretations shown in Figures 4 and 5

```
e1 = mobile-inside-zone-86
e2 = mobile-stopped-90
e3 = mobile-inside-zone-131
e4 = mobile-inside-zone-155
est = estimated event
```

Evidence	Time	Interpretation 4	Ranking 4	Interpretation 6	Ranking 6
e1	17:10:31	GPU-Enters..	100	GPU-Enters..	100
e2	17:10:32	GPU-Stopped..	1154	GPU-Stopped..	1154
e3	17:13:35	Airpl.-Enters-ERA	4885	Clutter	1154
e4	17:20:35	Clutter	4885	Airpl.-Enters-ERA	6143
est	17:20:35	Airpl.-Stopped..	194898		
est	17:21:35	Stop-Beacon	4489092		
est	17:27:35			Airpl.-Stopped..	245082
est	17:28:35			Stop-Beacon	5644964

The ratings for the partial interpretations of both alternatives are shown in Table 3. Interpretation 4 is the erroneous and Interpretation 6 the correct one. Initially, the arrival of the GPU sets a context where a vehicle is expected to enter the ERA, hence the crossing tanker is a candidate. But as soon as the true airplane enters, an alternative arises and is favoured because the probabilistic model expects an `Airplane-Enters-ERA` event 8 minutes after `GPU-Enters-GPU-Zone-Eve`, and the airplane's arrival is closer to that estimate than the tanker's. Note that clutter events not assigned to either of the two interpretations are not shown in the table.

The table also includes the estimated times of the next events `Airplane-Stopped-Inside-ERA` and `Stop-Beacon` together with the expected ratings of the competing interpretations. Considering that `Stop-Beacon` will occur after the true aircraft arrival and not at the time expected in Interpretation 4, the rating of this interpretation will surely be much lower than the estimated value, further increasing the distance between the right and the wrong interpretation.

Our experiments with concrete data have just begun, and we expect further interesting interpretations in the near future. However, it is safe to say that the probabilistic temporal model alone will not suffice to clearly separate good from bad interpretations, if the low-level data are very noisy. Another insight regards the quality of the model. If the model does not sufficiently match the ground truth, the ranking will be bad and false alternatives may win.

4. Summary and Outlook

We have described a novel probabilistic framework for real-time interpretation of multi-object scenes. It is based on the expectation that, as a scene evolves, several alternative interpretations may be possible initially and must be maintained in parallel. We have proposed a beam search paradigm where a limited number of alternatives is kept based on a probabilistic ranking. For domains with a hierarchical compositional structure, the probabilistic model can be realised as a Bayesian Compositional Hierarchy (BCH) which allows efficient updating for the incremental computation of ratings and for predictions of future events. An operational scene interpretation system called SCENIOR has been implemented which performs beam search guided by a BCH modelling the temporal relations of aircraft service activities. First results have been presented demonstrating the feasibility of the approach.

The work will be extended into several directions. First, more turnaround scenes will be interpreted and analysed to better tune the probabilistic model. Unfortunately, it cannot be expected to automatically learn a model for lack of a database with sufficiently many annotated scenes. Second, the characteristics of low-level errors will be analysed. So far, we are aware of many wrong classifications of the vehicle types (in our example, a tanker was mistaken for an airplane), of lost tracks at occlusions and of uncertain zone positions. Our basic approach to low-level uncertainty is to allow alternative interpretations, for example for ambiguous type classifications. But from our experiments we know that the number of parallel interpretation threads should stay below about 50 to guarantee real-time computer performance. This can only be achieved by judicious ranking and discarding of low-ranking alternatives.

References

[1] Tsotsos, J.K., Mylopoulos, J., Covvey, H.D., Zucker, S.W.: A Framework for Visual Motion Understanding. IEEE PAMI-2, 563-573 (1980)

[2] Nagel, H.-H.: From Image Sequences towards Conceptual Descriptions. Image and Vision Computing 6(2), 59-74 (1988)

[3] Neumann, B.: Description of Time-Varying Scenes. In: Waltz, D. (ed.), Semantic Structures, Lawrence Erlbaum, 167-206 (1989)

[4] Georis, B., Mazière, M., Brémond, F., Thonnat, M.: Evaluation and Knowledge Representation Formalisms to Improve Video Understanding. In: Proc. IEEE International Conf. on Computer Vision Systems ICVS06, IEEE Computer Society, 27 (2006)

[5] Neumann, B., Moeller, R.: On Scene Interpretation with Description Logics. In: Cognitive Vision Systems, Springer, LNCS 3948, 247–275 (2006)

[6] Rimey, R.D. Control of Selective Perception using Bayes Nets and Decision Theory. TR 468, Univ. of Rochester, Computer Science Department, Rochester, USA 14627 (1993)

[7] Koller, D., Pfeffer, A.: Object-oriented Bayesian Networks. In: The Thirteenth Annual Conference on Uncertainty in Artificial Intelligence, 302–313 (1997)

[8] Getoor, L., Taskar, B.: Introduction to Statistical Relational Learning (eds.), 129–174. The MIT Press (2007)

[9] Gyftodimos, E., Flach, P.A.: Hierarchical Bayesian Networks: A Probabilistic Reasoning Model for Structured Domains. In: de Jong, E., Oates, T. (eds.), Proc. Workshop on Development of Representations, ICML, 23–30 (2002)

[10] Neumann, B.: Bayesian Compositional Hierarchies - A Probabilistic Structure for Scene Interpretation. TR FBI-HH-B-282/08, Univ. of Hamburg, Department Informatik (2008)

[11] Mumford, D., Zhu, S.-C.: A Stochastic Grammar of Images. Now Publishers (2007)

[12] Lowerre, B.: The Harpy Speech Recognition System. Ph.D. thesis, Carnegie Mellon University (1976)

[13] Bohlken, W., Neumann, B.: Generation of Rules from Ontologies for High-level Scene Interpretation. In: G. Governatori et al. (eds.): Rule Interchange and Applications, Proc. International Symposium RuleML 2009, Springer LNCS 5858, 2009, 93-107