Visual Explanation by High-Level Abduction

On Answer-Set Programming Driven Reasoning about Moving Objects

Jakob Suchan¹, Mehul Bhatt¹-², Przemysław Wałega³, and Carl Schultz⁴

Cognitive Vision – www.cognitive-vision.org
EASE CRC – http://ease-crc.org

¹HCC Lab., University of Bremen, Germany, ²MPI Lab., Örebro University, Sweden
³University of Warsaw, Poland, and ⁴Aarhus University, Denmark

Abstract

We propose a hybrid architecture for systematically computing robust visual explanation(s) encompassing hypothesis formation, belief revision, and default reasoning with video data. The architecture consists of two tightly integrated synergistic components: (1) (functional) answer set programming based abductive reasoning with SPACE-TIME TRACKLETS as native entities; and (2) a visual processing pipeline for detection based object tracking and motion analysis.

We present the formal framework, its general implementation as a (declarative) method in answer set programming, and an example application and evaluation based on two diverse video datasets: the MOTChallenge benchmark developed by the vision community, and a recently developed Movie Dataset.

Introduction

A range of empirical research areas such as cognitive psychology and visual perception articulate human visual sense-making as an inherently abductive (reasoning) process [Moriarty, 1996; Magnani, 2015] involving tight linkages between low-level sub-symbolic processes on the one hand, and high-level object and event-based segmentation and inference involving concepts and relations on the other. In spite of the state of the art in artificial intelligence and computer vision, and most recent advances in neural visual processing, generalised explainable visual perception with conceptual categories in the context of dynamic visuo-spatial imagery remains an exceptionally challenging problem presenting many research opportunities at the interface of Logic, Language, and Computer Vision.

Explainable Visual Perception

We define explainable visual perception from a human-centred, and commonsense reasoning viewpoint. In this paper, it denotes the ability to declaratively:

\( V\!\!X\!P_1 \): hypothesise spatio-temporal belief (states) and events; events may be both primitive or temporally-ordered aggregates; from a more foundational viewpoint, what is alluded to here is a robust mechanism for counterfactual reasoning.

\( V\!\!X\!\!P_2 \): revise spatio-temporal beliefs, e.g., by non-monotonically updating conflicting knowledge, to fix inherently incompatible configurations in space-time defying geometric constraints and commonsense laws of naive physics, e.g., pertaining to physical (un)realisability, spatio-temporal continuity.

\( V\!\!X\!\!P_3 \): make default assumptions, e.g., about spatio-temporal property persistence concerning occupancy or position of objects; identity of tracked objects in space-time.

Explanatory reasoning in general is one of the hallmarks of general human reasoning ability; robust explainable visual perception particularly stands out as a foundational functional capability within the human visuo-spatial perception faculty. In this respect, the following considerations — establishing the scope of this paper— are important wrt. \( V\!\!X\!\!P_{1-3} \):

- our notion of explainability is driven by the ability to support commonsense, semantic question-answering over dynamic visuo-spatial imagery within a declarative KR setting;
- the features alluded to in \( V\!\!X\!\!P_{1-3} \) are not exhaustive; we focus on those aspects that we deem most essential for the particular case of movement tracking.

A Hybrid Architecture for Visual Explanation

This paper is driven by the development of a visual explanation component within a large-scale computational vision & perception system targeted at a range of cognitive interaction technologies and autonomous systems where dynamic visuo-spatial imagery is inherent.

The key contribution is a hybrid visual explanation method based on the integration of high-level abductive reasoning within Answer Set Programming (ASP) [Brewka et al. (2011)] on the one hand, and low-level visual processing for object tracking on the other. The core focus of the paper is on the theory, implementation, and applied evaluation of the visual explanation method. We particularly emphasise the closely-knit nature of two key sub-components representing abductive explanation (\( \Sigma_{abd} \)) and low-level motion tracking (\( \Sigma_{etr} \)) modules respectively:

\( \Sigma_{abd} \). ASP-based abductive reasoning with abstract visuo-spatial concepts — such as OBJECTS, EVENTS, SPACE-TIME TRACKLETS — as native objects within ASP.
\( \Sigma_{trk} \) : Low-level visual processing pipeline for motion tracking, consisting of detection-based object tracking and optical-flow based (scene-level) movement tracking

The abductive component \( \Sigma_{abd} \) is suited for a wide-range of dynamic visuo-spatial imagery; however, we only focus on video in this paper. As an application, we focus on scene interpretation from video with two datasets: a Movie Dataset [Suchan and Bhatt, 2016a] and the MOT16 Dataset, a benchmark dataset released as part of The Multiple Object Tracking Challenge [Milan et al., 2016a].

**Visual Explanation: A Hybrid Architecture**

We present a general theory for explaining visuo-spatial observations by integrating low-level visual processing and high-level abductive reasoning (Fig. 1). As such, we consider visual abduction as reasoning from visual observations to explanations consisting high-level events grounded in low-level motion tracks. The resulting set of hypotheses is optimised based on the abduced events and the corresponding object movement.

**Ontology: Space, Time, Objects, Events**

The framework for abducting visual explanations is based on visuo-spatial domain objects representing the visual elements in the scene. The domain objects are associated with spatio-temporal objects describing motion tracks obtained from \( \Sigma_{trk} \), which form the basis for qualitative spatio-temporal abstractions facilitating high-level reasoning about visuo-spatial dynamics.

The *Qualitative Spatio-Temporal Domain* (QS) is characterised by the basic spatial and temporal entities (\( \mathcal{E} \)) that can be used as abstract representations of domain-objects and the relational spatio-temporal structure (\( \mathcal{R} \)) that characterises the qualitative spatio-temporal relationships amongst the supported entities in (\( \mathcal{E} \)). For this paper, we restrict the basic spatial entities to:

- **points** are a pair of reals \( x, y \),
- **axis-aligned rectangles** are a point \( p \) and its width and height \( w, h \),

and the temporal entities to:

- **time-points** are a real \( t \)

Visuo-spatial domain objects \( \mathcal{O} = \{ o_1, o_2, \ldots, o_n \} \) are described as spatio-temporal objects by a set of spatial entities, i.e., **points**, and **axis-aligned rectangles**, in time. Towards this, \( \mathcal{MT} \) contains all object tracks obtained form \( \Sigma_{trk} \).

The track of a single object \( o_i \) is represented by \( \mathcal{MT}_{o_i} = (\varepsilon_{t_1}, \ldots, \varepsilon_{t_e}) \), where \( t_1 \) and \( t_e \) denote the start and end frame of the track and \( \varepsilon_{t} \) denotes a spatial primitive representing the object \( o_i \) at the time points \( t_a \) to \( t_e \), e.g., the axis aligned bounding box of the object.

For reasoning about visuo-spatial phenomena of object tracks, spatio-temporal relationships (\( \mathcal{R} \)) between the basic entities in \( \mathcal{E} \) may be characterised with respect to arbitrary spatial and spatio-temporal domains such as merotopy, orientation, distance, size, motion. From the viewpoint of the examples of this paper, it suffices to focus on the language of the merotopological system of the Region Connection Calculus (RCC8) [Randell et al., 1992] consisting of the following jointly exhaustive and pair-wise disjoint relations: disconnected (dc), externally connected (ec), partially overlapping (po), equal (eq), (non-) tangential proper part (\((n)pp\)), and their inverse (\((n)tp\)).

Abducable events (\( \Theta \)) and beliefs (\( \Phi \)) are defined by their (spatio-temporal) **preconditions** and **observer effects**, i.e., for each event \( \theta \in \Theta \) we define which properties of the scene have to be true for the event to be possible, and what the (visible) effects of the event are. In the case of visual abduction, properties of the scene are determined by the visually observed object tracks and represent qualitative relations between tracks, i.e., spatial relation \( r \in \mathcal{R} \) holding between basic spatial entities \( \varepsilon \) of a motion track. Complex events are defined by combining multiple events and beliefs, e.g., an event of an object \( o_i \) passing behind another object \( o_j \) can be defined based on the events of \( o_i \) being occluded by \( o_j \) and \( o_i \) and \( o_j \) changing sides.

**Abducting Visual Explanations**

We implement the theory for visual explanations combining visual processing for object detection and tracking, and estimating movements in the scene, with ASP based reasoning about events, objects, and spatial-dynamics (Fig. 1). The main components of the overall tightly-integrated system comprising of low-level motion tracking with high-level explanation is as follows:
I. Visuo-Spatial Observations (VO) — low-level visual processing consisting of detection based tracking of object and people movements.

II. Hypotheses (H) — abducting hypotheses including belief states, events, and default assumptions given a set of visuo-spatial observations (VO).

III. Hypotheses to Explanations — as encompassed in $\forall X P_{1-3}$ — are generated by evaluating abduced hypothesis (H) based on high-level optimisation of event sequences and low-level cost minimisation of corresponding motion tracks.

I. Visuo-Spatial Observations (VO) — Visual explanations are based on observations obtained from visuo-spatial imagery, e.g., video, RGB-D. For the examples in this paper, we focus on detection and tracking of people and objects for estimating motion trajectories of semantic entities in the scene. However, the presented approach is also capable of incorporating other kinds of motion, e.g., optical flow based low-level movement analysis using long term observations [Ochs et al., 2014], or dense motion-tracks [Gaidon et al., 2014] for estimating pixel level motion, corresponding to camera movement, or fine grained object motion, etc. This may be used to abduce fine grained interactions and the interplay of different movements, e.g. people movement in the presence of camera movement, by combining motion trajectories of semantic entities with pixel movements.

Movement of people and objects is estimated following the tracking by detection paradigm, which is based on object detections for each frame and association of the detections across frames. Object detections can in principle be obtained using any state of the art (deep learning based) detector (e.g., faster RCNN [Ren et al., 2015], YOLO [Redmon et al., 2016]), or deformable part models (DPM) [Felzenszwalb et al., 2010]. For the examples in this paper we are using faster RCNN in the movie examples and DPM detections for the MOT dataset (which come as part of the dataset). For association of detections we apply the well established approach of combining min cost assignment and kalman filters for finding optimal tracklets, where the cost for assigning a detection to a track is calculated by the distance between the prediction for a track and the detection.

• Prediction — for each track Kalman filters are used to predict the next position of the track and the costs for each detection is calculated based on the distance between the prediction and the detection.

• Assignment — detections are assigned to a track using min cost assignment which calculates the best assignment of detection to tracks based on the costs calculated in the prediction step. If no assignment is possible for a detection a new track is started.

The resulting object tracks $MT$ form the basis for abducting explanations on movement events occurring in the input data.

II. Hypotheses (H) — Explanations for visual observations are abduced based on a sequence of visual observations obtained from the video data. For abducting visual explanations from VO, given,

- set $VO$ consisting of visuo-spatial observations obtained from $\Sigma_{trk}$,
- domain independent theory of space and time ($\Sigma_{space}$) based on the spatio-temporal ontology ($QS$)
- observable events ($\Sigma_{events}$)
- domain dependent background knowledge, describing properties of the domain ($\Sigma_{domain}$)

the task of visual abduction is to find a set of logically consistent hypotheses $H$ consisting of high-level events and beliefs grounded in low-level motion tracks, such that:

$$\Sigma_{space} \land \Sigma_{events} \land \Sigma_{domain} \land H \models VO$$

The computed hypotheses (H) are based on abducibles constituting primitive events and beliefs: $H \equiv H_{Events} \land H_{Belief}$; these hypotheses in turn are directly usable for inducing motion tracks:

$$MT_{VXP} \leftarrow H_{event} \land H_{belief} \land MT$$

The resulting motion tracks $MT_{VXP}$ represent the low-level instantiation of the abduced high-level event sequence.

III. Hypotheses to Explanations — Hypotheses for visual observations (VO) may be ranked based on the abduced event sequences and cost minimisation of corresponding motion trajectories, i.e., the costs for connecting motion tracks in the hypothesised movements, e.g., considering changes in velocity, size, and length of missing detections. As such, hypothesised explanations are ranked using the built in optimisation functionality of ASP1. In particular, we use minimisation by assigning preferences to the abducibles events and beliefs and optimise towards minimising the costs of events and beliefs in the answer. E.g., by minimising the duration of missing detections for a particular object, or minimising assigning the property noise to a track to explain its observation.

• High-level event sequences — the cost for high-level events is estimated by assigning a cost for each event. Additionally, for events having a duration there is also a cost assigned to the length of the event, e.g., to abduce that a track is noise is more likely, when it is a very short track. These costs are weighted based on the abduced event that caused the missing detections, e.g., missing detections caused by an occlusion are more likely to be longer (and therefore have a lower cost), than missing detections caused by the detector.

1For optimisation we use ASP with the so-called weak constraints (Gebser et al. [2012]), i.e., constraints whose violation has a predefined cost. When solving an ASP program with weak constraints, a search for an answer set with a minimal cost of violated constraints is performed. Each such minimal-cost answer set is called optimal. The mechanism involving weak constraints enables us to set preferences among hypothesised explanations and search for the ones that are most preferred (optimal). Importantly, the approach enables us to exhaustively search for all optimal explanations. As a result, we can subsequently use other (more fine-graded) evaluation techniques to choose the most preferred explanations.


| EVENTS | Description |
|-----------------|-------------|
| enters(Border, Trk, T) | The object corresponding to track Trk enters the scene at time point T. |
| exits(Border, Trk, T) | The object corresponding to track Trk exits the scene at time point T. |
| occludes(Trk1, Trk2, Trk3, T1, T2) | The object corresponding to track Trk1 and track Trk2 is occluded by the object corresponding to track Trk3 between time points T1 and T2. |
| missing_det(Trk1, Trk2, T1, T2) | Missing detections for the object corresponding to the tracks Trk1 and Trk2 between time points T1 and T2. |

| COMPLEX EVENTS | Description |
|----------------|-------------|
| passing_behind(O1, O2, T1, T2) | Object O1 is passing behind object O2 between time points T1 and T2. |
| moving_together(O1, O2, T1, T2) | Objects O1 and O2 are moving together between time points T1 and T2. |

| BELIEFS | Description |
|-----------------|-------------|
| same_object(Trk1, Trk2) | The tracks Trk1 and Trk2 belong to the same object. |
| belongs_to(Trk, Trk1) | The object corresponding to track Trk1 is a part of the object corresponding to track Trk. |
| noise(Trk) | Track Trk is a faulty detection. |

Table 1: Abducibles: Events and Beliefs for Explaining Observed Object Tracks.

- **Low-level motion characteristics** the cost of the motion tracks $\mathcal{M}$ is estimated based on the characteristics of the abduced movement. Towards this we consider changes in velocity, for each abduced event that connects two object tracks. For the examples in this paper we use a constant velocity model to minimize changes in velocity of an abduced object track.

The best explanation is selected by *minimising* the costs of the hypothesised answer set based on the motion and the high-level event sequence. The final movement tracks for the optimal explanation $\mathcal{M}_{\mathcal{V}_{XP}}$ are then generated by predicting the motion of the object for each hypothesised event associating two tracks, using linear interpolation.

**Visuo-Spatial Phenomena**

The framework may be used for abducting explanations by modelling visuo-spatial phenomena including but not limited to:

- **Object Persistence** objects can not appear and disappear without a cause, e.g. getting occluded, leaving the field of view of the camera, etc.
- **Occlusion** objects may disappear or re-appear as a result of occlusion between two non-opaque objects.
- **Linkage** objects linked to each other, such that movement of one object influences movement of the other object, e.g. a face belonging to a person.
- **Sensor Noise** observations that are based on faulty data, e.g. missing information, miss-detections, etc.

**Event Semantics as Spatial Constraints** For explaining perceived visuo-spatial dynamics of objects in the scene, we define the basic events listed in Table 1 to assure spatio-temporal consistency, e.g. object persistence, or occlusion. The focus is on explaining appearance and disappearance of objects in the scene.

- **Entering and Leaving** Objects can only enter or exit the scene by leaving the screen at one of its borders. For these events to happen the object has to be overlapping with the border of the screen while appearing or disappearing.

```
enters:
  topology(po,TRbox,left_border) :-
  enters(from_left,TR,T), track(TR,TRbox,T).

enters:
  topology(po,TRbox,right_border) :-
  enters(from_right,TR,T), track(TR,TRbox,T).
```

- **Missing Detecions and Occlusion** Appearance and disappearance of tracks in the middle of the screen can be either caused by a missing detection or by an occlusion from some other object. The event that an object gets occluded by some other object may be possible, when the object disappears while overlapping with the other object.

```
occludes:
  topology(po,TRbox1,TRbox2) :-
  occludes(TR1,TR2a,TR2b,T1,T2),
  track(TR1,TRbox1,T1), track(TR2a,TRbox2,T1).

occludes:
  topology(po,TRbox1,TRbox2) :-
  occludes(TR1,TR2a,TR2b,T1,T2),
  track(TR1,TRbox1,T2), track(TR2b,TRbox2,T2).
```

**Generating Hypotheses on Events** We generate hypotheses explaining the observation of a track starting and ending based on the defined events, such that the spatial constraints defined above are satisfied.

```
starts:
  l:
  noise(TR);
  enters(from_left,TR,T1);
  enters(from_right,TR,T);
  missing_det(TR1,TR,T1,T2) : ends(TR1,T1), T1<T;
  occludes(TR1,TR2a,TR2b,T1,T2) : ends(TR2a,T1), T1<T,
  type(TR1,Type), Type=border, starts(TR1,T1),
  ends(TR1,T2), T1<T<T2
  )
  :- starts(TR,T).
```

\(^2\)The semantics of the underlying spatial and temporal relations with (QS) is founded on the geometric and spatial reasoning capability provided by the ASPMT(QS) spatial reasoning system [Walega et al., 2015]; the system, implemented within ASPMT [Lee and Meng, 2013], is directly available to be used as a black-box within our visual explanation framework.
Beliefs as (Spatial) Constraints  Beliefs about objects in the scene are stated as constraints in ASP.

- **Part-Whole Relations**  E.g. the fact that every face belongs to exactly one person is stated as follows.

  ```prolog
  :- belongs_to(Face1, Person),
     belongs_to(Face2, Person),
     Face1 != Face2.
  ```

Further we define that the face of a person has to stay together with the person it belongs to, using spatial constraints, i.e. the face track is a non-tangential proper part of the person track.

```prolog
topology(tpp, Pbox, Pbox) :-
  belongs_to(Person, Face),
  track(Person, Pbox, N),
  track(Face, Pbox, N).
```

Generating Hypotheses on Beliefs  Hypotheses on faces belonging to persons are generated by stating that for each detected face, there has to be a corresponding person, such that the spatial constraint is satisfied.

```prolog
1{ belongs_to(Face, Person) : type(Person, person) }.
:- type(Face, face).
```

Costs of Hypotheses using Optimization  Costs for abduced visual explanations are minimized using ASP based optimization, e.g., the cost for missing detections are based on their length.

```prolog
#minimize { (T2_start-T1_end) * ALPHA, TR1, TR2 :
  missing_det(TR1, TR2, T1_end, T2_start),
  weight(missing_det, ALPHA) }.
```

Further, the characteristics of the underlying motion is taken into account, assuming constant velocity, by taking differences in velocity between the two tracks and the interpolated segment in between.

```prolog
#minimize { ((X_vel_prev-2 * X_vel_during * X_vel_next)**2 +
  (Y_vel_prev-2 * Y_vel_during * Y_vel_next)**2) * ALPHA,
  TR1, TR2 :
  missing_det(TR1, TR2, T1, T2),
  track(TR1, Box_T1, T1),
  box(Box_T1, X_e, Y_e, __, __),
  [... long ...]
  X_vel_during = (X_e - X_s) / (T2-T1),
  Y_vel_during = (Y_e - Y_s) / (T2-T1),
  [... long ...]
  weight(missing_det_vel, ALPHA) }.
```

Application and Evaluation: Scene Interpretation with Moving Objects

We demonstrate the proposed theory of visual abduction by applying it in the context of scene interpretation focussing on generating visual explanations on perceived motion. In particular, the emphasis is on spatio-temporal consistency of abduced explanations with respect to the underlying motion tracks.

Movie Dataset  [Suchan and Bhatt, 2016a,b].  We use the video part of the Movie Dataset consisting of 16 select scenes from 12 films, with each scene ranging between 0 : 38 minute to max. of 9 : 44 minutes in duration. Most of the scenes involve multiple moving objects, and moving camera(s). Object detection with the movie dataset is performed using faster RCNN [Ren et al., 2015] with the pre-trained VGG16 model for detection of people and objects in the scene.

Visual Explanation of Object Movement  As an example consider the scene from the movie The Grand Budapest Hotel (2014) by Wes Anderson (Figure 2). Here we abduce the movement of the two main characters walking down the hallway of the hotel. The set of visual observations consist of 11 tracks for the detected people in the scene. The abduced events explain occuring missing detections, occlusion and re-appearance, as well as entering, and leaving the scene.

```prolog
exits(to_right, trk18, 1511) enters(from_left, trk9, 1500)
occludes(trk4, trk2, trk8, 1496, 1495)
occludes(trk1, trk6, trk7, 1496, 1496)
...
noise(trk4) noise(trk8) ending(trk11, 1512)
```
Visual Explanation by High-Level Abduction
On Answer-Set Programming Driven Reasoning about Moving Objects
Jakob...exits(to_right,trk25,47) occludes(trk1,trk23,trk28,50,54)

Similarly we can abduce complex events based on movement events and beliefs, e.g., based on the following sequence of movement events from the movie “The Bad Sleep Well” (1960) by Akira Kurosawa (depicted in Figure 3), we can abduce the occurrence of the complex event passing behind between two objects in the scene.

... missing_det(trk1,trk3,556,561) noise(trk2) occlusion(trk3,trk4,trk0,561,629) missing_det(trk4,trk5,629,634) ...

... passing_behind(obj1,obj2,561,629) ...

Hypotheses on People and their Faces As an example for abducting properties of objects, we use face detections and abduce which face belongs to which person, using the part-whole relations defined in Section Visuo-Spatial Phenomena. The spatial constraint that a face track has to be inside a person track, is used to improve abduced object tracks.

MOT16 Benchmark Video Dataset We use the MOT16 [Milan et al., 2016a] dataset consisting of highly accurate and consistent annotation protocols. MOT16 is a benchmark dataset released as part of The Multiple Object Tracking Challenge (MOTChallenge). It consists of 14 complex video sequences in highly unconstrained environments filmed with both static and moving cameras. We use the detections (provided by the dataset) based on deformable part models (DPM): these are noisy and include numerous detection errors, i.e. false positives and false negatives. We focus on abducing people motion and on generating concise explanations for the perceived movements, i.e. under consideration of occlusion and appearance / disappearance as per the abducible events in Table 1. As a result of the noisy detections and the complexity of the movements in the dataset the obtained motion tracks include a high amount of errors, e.g. identity switches, missing detections, etc. (Figure 4). For the sample scene we abduced the following events:

... missing_det(trk19,trk23,45,49) exits(to_right,trk25,47) exits(to_right,trk6,46) occludes(trk1,trk23,trk28,50,54) ...

Evaluating Visual Explanations
We evaluate the generated visual explanations based on their ability to generate low-level object tracks. Towards this we compare the accuracy and precision of the movement tracks the hypothesised event sequences are grounded in.

Multi-Object Tracking For evaluating the precision and accuracy of the abduced object tracks we follow the ClearMOT evaluation schema for evaluating multi-object tracking performance as described in [Bernardin and Stiefelhagen, 2008].

- MOTA describes the accuracy of the tracking, taking into account the number of missed objects / false negatives (FN), the number of false positives (FP), and the number of miss-matches (MM).
- MOTP describes the precision of the tracking based on the distance of the hypothesised track to the ground truth of the object it is associated to.

These metrics are used to assess how well the generated visual explanations describe the low-level motion in the scene.

Results & Discussion
We present results of the presented approach for abducting visual explanations (VXP) on improving multi-object tracking performance using selected scenes from the Movie Dataset and the MOT 2016 Dataset. Overall the results show that using our proposed method can increase accuracy (MOTA) of the tracking. However, the precision (MOTP) of the tracking is dropping a little, which is a result of the interpolation, which is not as precise as the detections.

Movie Dataset The scenes in the Movie Dataset contain relatively controlled scenes with few targets. Results on these scenes show that the presented approach can abduce correct event sequences and is capable of correcting many of the errors normally occurring in multi-object tracking tasks, e.g., fragmented object tracks, id-switches, etc. I.e., the object tracks obtained from the high-level event sequences improve the accuracy (MOTA) of the tracking (see Table 2).

MOT Dataset The results for the visual tracking on the Venice-2 file from the MOT2016 dataset (see Table 2) show, that our approach is capable of dealing with complex data in challenging settings. For comparability we use the DPM
based detections provided with the dataset for our evaluation. These detections suffer from a large number of false positives and negatives. Due to this our underlying tracking method is only capable of tracking a small part of the overall targets in the data, resulting in a low MOTA score. Even so our results demonstrate that high-level abduction can be used to improve tracking performance, i.e., improved MOTA by 1.7%, and number of miss-matches reduced by 43.8%.

Based on the promising results presented in this paper, using basic tracking by detection, we suppose that ASP-based visual explanations can also be used to improve multi-object tracking using more elaborate tracking approaches, e.g., based on continuous energy minimization [Milan et al., 2016b] or minimum cost multi-cuts [Tang et al., 2017]. For the examples presented in this paper, optimal answer-sets are computed rather fast, e.g., for the scene depicted in Figure 2, 457 optimal models are abduced in 0.973s, of which the first model is found after 0.04s and the last one after 0.93s. For longer scenes or in online situations, visual explanations would naturally have to be computed incrementally, as the number of abduced hypothesis grows exponentially with the number of tracks.

### Related Work

Recently, Answer Set Programming (ASP) has become a widely used tool for abductive reasoning and non-monotonic reasoning in general. The work presented in this paper aims at bridging the gap between high-level formalisms for logical abduction and low level visual processing, by tightly integrating qualitative abstractions of space and time with the underlying numerical representations of spatial change. The significance of abducing high-level explanations in a range of contexts has been well established in AI and KR, e.g. in planning and process recognition [Kautz and Allen, 1986; Kautz, 1991], vision and abduction [Shanahan, 2005], probabilistic abduction [Blythe et al., 2011] etc. Within KR, reasoning about spatio-temporal dynamics on the basis of an integrated theory of space, time, objects, and position [Galton, 2000] or defined continuous change using 4-dimensional regions in space-time has also received significant theoretical interest [Muller, 1998; Hazarika and Cohn, 2002]. Dubba et al. [2015] uses abductive reasoning for improving learning of events in an inductive-abductive loop, using inductive logic programming (ILP). The role of visual commonsense

---

### Table 2: Evaluation of Tracking Performance

| Sequence               | Tracking  | MOTA   | MOTP   | FP | M  | MM  | non-r. MM | r. MM | TP | TR |
|------------------------|-----------|--------|--------|----|----|-----|-----------|-------|----|----|
| The Bad Sleep Well     | without VXP | 58.5 % | 80.8 % | 1  | 86 | 5   | 0        | 5     | 0.875 | 1.0 |
| (107 frames, 2 targets)| with VXP   | 100.0% | 69.1%  | 0  | 0  | 0   | 0        | 0     | 1.0 | 1.0 |
| The Drive              | without VXP | 59.8%  | 76.7%  | 0  | 345| 18  | 0        | 18    | 1.0 | 1.0 |
| (627 frames, 2 targets)| with VXP   | 79.7%  | 76.6%  | 0  | 182| 1   | 0        | 1     | 1.0 | 1.0 |
| MOT2016 - Venice-2     | without VXP | 6.4%   | 69.9%  | 47 | 27137| 153 | 4        | 150   | 0.987 | 0.241 |
| (600 frames, 74 targets)| with VXP   | 8.1%   | 65.2%  | 216| 26535| 86  | 27       | 77    | 0.946 | 0.486 |

in general, and answer set programming in particular, has been used in conjunction with computer vision to formalise general rules for image interpretation in the recent works of Aditya et al. [2015]. From the viewpoint of computer vision research there has been an interest to synergise with cognitively motivated methods [Aloimonos and Fermüller, 2015]; in particular the research on semantic interpretation of visual imagery is relevant to this paper, e.g., for combining information from video analysis with textual information for understanding events and answering queries about video data [Tu et al., 2014], and perceptual grounding and inference [Yu et al., 2015].

### Summary and Outlook

The paper presents a robust, declarative, and generally usable hybrid architecture for computing visual explanations with video data. With a focus on abductive reasoning in the context of motion tracking, the architecture has been formalised, fully implemented, evaluated with two diverse datasets: firstly, the benchmark MOTChallenge (evaluation focus), and secondly a Movie Dataset (demonstration focus).

The overall agenda of the work in this paper is driven by a tighter integration of methods in KR and Computer Vision on the one hand, and the twin concepts of “deep semantics” & “explainability” on the other. VXP is rooted in state of the art methods in knowledge representation and reasoning (i.e., answer set programming), and computer vision (detection based object tracking, optical flows, RCNN). The overall system is designed to be a part of a larger perception module within autonomous systems, and cognitive interaction systems. The scope of VXP may be further expanded, e.g., for visuo-spatial learning (with inductive logic programming), ontological reasoning (with description logics), are achievable depending on the scope and complexity of the low-level visual signal processing pipeline, and chosen high-level commonsense knowledge representation and reasoning method(s) at hand.

### References

Somak Aditya, Yezhou Yang, Chitta Baral, Cornelia Fermüller, and Yiannis Aloimonos. Visual commonsense for scene understanding using perception, semantic parsing and reasoning. In 2015 AAAI Spring Symposium Series, 2015.

Yiannis Aloimonos and Cornelia Fermüller. The cognitive dialogue: A new model for vision implementing common sense reasoning. Image and Vision Computing, 34:42 – 44, 2015.
Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008(1):246309, May 2008.

Mehul Bhatt and Seng W. Loke. Modelling dynamic spatial systems in the situation calculus. *Spatial Cognition & Computation*, 8(1-2):86–130, 2008.

Mehul Bhatt, Hans W. Guesgen, Stefan Wölfl, and Shyamanta Moni Hazarika. Qualitative spatial and temporal reasoning: Emerging applications, trends, and directions. *Spatial Cognition & Computation*, 11(1):1–14, 2011.

Mehul Bhatt. Reasoning about space, actions and change: A paradigm for applications of spatial reasoning. In *Qualitative Spatial Representation and Reasoning: Trends and Future Directions*. IGI Global, USA, 2012.

James Blythe, Jerry R. Hobbs, Pedro Domingos, Rohit J. Kate, and Raymond J. Mooney. Implementing weighted abduction in markov logic. In *Proceedings of the Ninth International Conference on Computational Semantics*, IWCS ’11, USA, 2011. Association for Computational Linguistics.

Gerhard Brewka, Thomas Eiter, and Miroslaw Truszczynski. Answer set programming at a glance. *Commun. ACM*, 54(12):92–103, December 2011.

Krishna Sandeep Reddy Dubba, Anthony G. Cohn, David C. Hogg, Mehul Bhatt, and Frank Dyilla. Learning relational event models from video. *J. Artif. Intell. Res. (JAIR)*, 53:41–90, 2015.

Pedro F. Felzenszwalb, Ross B. Girshick, David A. McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, 2010.

Adrien Gaidon, Zaïd Harchaoui, and Cordelia Schmid. Activity representation with motion hierarchies. *International Journal of Computer Vision*, 107(3):219–238, 2014.

Anthony Galton. *Qualitative Spatial Change*. Oxford University Press, 2000.

M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. *Answer Set Solving in Practice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers, 2012.

Shyamanta M Hazarika and Anthony G Cohn. Abducing qualitative spatio-temporal histories from partial observations. In *KR’98. Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann, Italy, 1998.

Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(6):1187–1200, 2014.

David A. Randell, Zhan Cui, and Anthony Cohn. A spatial logic based on regions and connection. In *KR’92. Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann, San Mateo, California, 1992.

Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, Las Vegas, NV, USA, June 27-30, 2016, pages 779–788. IEEE Computer Society, 2016.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Annual Conference on Neural Information Processing Systems 2015*, Canada, 2015.

Murray Shanahan. Perception as abduction: Turning sensor data into meaningful representation. *Cognitive Science*, 29(1):103–134, 2005.

Jakob Suchan and Mehul Bhatt. The geometry of a scene: On deep semantics for visual perception driven cognitive film, studies. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2016*, USA. IEEE Computer Society, 2016.

Jakob Suchan and Mehul Bhatt. Semantic question-answering with video and eye-tracking data: AI foundations for human visual perception driven cognitive film studies. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI 2016*, New York, USA. IJCAI/AAAI Press, 2016.

Jakob Suchan and Mehul Bhatt. Deep semantic abstractions of everyday human activities - on commonsense representations of human interactions. In *ROBOT 2017: Third Iberian Robotics Conference, Seville, Spain*. Springer, 2017.

Siyu Tang, Mykhaylo Andriluka, Bjorn Andres, and Bernt Schiele. Multiple people tracking by lifted multicat and person re-identification. In *CVPR*, 2017.

Kewei Tu, Meng Meng, Mun Wai Lee, Tae Eun Choe, and Song Chun Zhu. Joint video and text parsing for understanding events and answering queries. *IEEE MultiMedia*, 2014.

Przemysław Wałęga, Mehul Bhatt, and Carl Schulz. ASPMT(QS): Non-Monotonic Spatial Reasoning with Answer Set Programming Modulo Theories. In *LPNMR: Logic Programming and Nonmonotonic Reasoning - 13th International Conference*, Lexington, KY, USA, 2015.

Haonian Yu, N. Siddharth, Andrei Barbu, and Jeffrey Mark Siskind. A Compositional Framework for Grounding Language Inference, Generation, and Acquisition in Video. *J. Artif. Intell. Res. (JAIR)*, 52:601–713, 2015.
Acknowledgements

We acknowledge funding as part of the German Research Foundation (DFG) CRC 1320 “EASE – Everyday Activity Science and Engineering” (http://www.ease-crc.org/) Project P3: “Spatial Reasoning in Everyday Activity”. We also acknowledge the Polish National Science Centre project 2016/23/N/HS1/02168.

SUPPLEMENTARY MATERIAL FOR:

J. Suchan., M. Bhatt, Walega, P., Schultz, C. (2018). Visual Explanation by High-Level Abduction: On Answer-Set Programming Driven Reasoning about Moving Objects. In AAAI 2018: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, February 2-7, 2018, New Orleans, USA.

CONTENTS

We have submitted the following supplementary material with the paper (S1 – S3):

S1. A Sample Input Data File
S2. Complete ASP Program
S3. Additional Examples from the Vision MOT Dataset etc (Figs. 5-7 in the supplementary part)

SUPPLEMENTARY 1:
INPUT DATA

Input Data This includes movement tracks, consisting of media properties, borders of the frame, and object tracks represented as boxes with x, y, width, and height, and the time t given by the frame number with the ASP Program.

%% Media Properties
frame_width(1280).
frame_height(720).

%% shot start and end
shot(327, 3983).

%% FOV borders
type(left_border, border).
box(left_border, 10, 360, 20, 360).
type(right_border, border).
box(right_border, 1270, 360, 20, 360).

%% TRACK: 1
track_time(trk1, 344, 346).
type(trk1, person).
box(bbox0_trk1, 693, 357, 1028, 602).
box(bbox1_trk1, 699, 364, 921, 621).
box(bbox2_trk1, 701, 367, 921, 621).
track(trk1, bbox0_trk1, 344).
track(trk1, bbox1_trk1, 345).
track(trk1, bbox2_trk1, 346).
...

%% TRACK: 0
track_time(trk0, 327, 419).
type(trk0, person).
box(bbox0_trk0, 346, 386, 470, 521).
...
box(bbox83_trk0, 579, 389, 426, 557).
track(trk0, bbox0_trk0, 327).
...
track(trk0, bbox83_trk0, 419).
...
**SUPPLEMENTARY 2: ANSWER SET PROGRAMMING BASED HYPOTHESES GENERATION**

Abducing Movement Events & Beliefs  ASP code for abducing events and beliefs on motion tracks.

Properties of the tracks, i.e., *start*, *end* and *duration*.

```
%% Start / End of Tracks
starts(TR, T_start) :- track_time(T, T_start, T_end).
ends(TR, T_end) :- track_time(T, T_start, T_end).
```

```
%% Duration of Tracks
duration(TR, T_dur) :-
  track_time(T, T_start, T_end),
  T_dur = T_end-T_start.
```

Choice rules for generating explanations on observed changes in the tracks, i.e., *starting* and *ending* of a track.

```
%% Generate hypothesis
1;[
  noise(TR):
    duration(TR, T_dur), T_dur < 10;
  beginning(TR, T) : shot(T,_,_);
  enters(from_left,TR,T);
  enters(from_right,TR,T);
  missing_det(TR1,TR,T1,T):
    ends(TR1,T1), T1<T, T-T1 < 50;
  missing_from_start(TR, T):
    shot(T,T_start,_, T-T_start < 20;
  occludes(TR1,TR2,TR,T1,T):
    ends(TR2,T1), T1<T, Type!="border",
    starts(TR1,T1), ends(TR1,T12),
    T11<T1, T<T12;
  no_exp_start(TR,T)
]
1;:- starts(TR,T).
```

```
1;[
  noise(TR):duration(TR, T_dur), T_dur < 10;
  ending(TR, T) : shot(_,T,_,);
  exits(to_left,TR,T);
  exits(to_right,TR,T);
  missing_det(TR2,TR,T2):
    starts(TR2,T2), T<T2, T2-T < 50;
  missing_till_end(TR, T):
    shot(_,T_end, T, T_end-T < 20;
  occludes(TR1,TR,TR2b,T1,T):
    starts(TR2b,T1), T1<T1,
    type(TR1,Type), Type!="border",
    starts(TR1,T11), ends(TR1,T12),
    T11<T, T11<T12;
  no_exp_end(TR,T)
]
1;:- ends(TR,T).
```

Spatial preconditions for abducible events using topological relations from the region connection calculus.

```
# entering

topology(po,TRbox, _left.Border) :-
  enters(from_left,TR,T), track(TR,TRbox,T).

topology(po,TRbox, _right.Border) :-
  enters(from_right,TR,T), track(TR,TRbox,T).
```

```
# exiting

topology(po,TRbox, _left.Border) :-
  exits(to_left,TR,T), track(TR,TRbox,T).

topology(po,TRbox, _right.Border) :-
  exits(to_right,TR,T), track(TR,TRbox,T).
```

```
# occluding

topology(po,TRbox1,TRbox2) :-
  occludes(TR1,TR2,TR,T1),
  track(TR1,TRbox1,T1), track(TR2,TRbox2,T1).

topology(po,TRbox1,TRbox2) :-
  occludes(TR1,TR2,TR,T1),
  track(TR1,TRbox1,T2), track(TR2,TRbox2,T2).
```

Multi-Objective Optimization  ASP based optimization of generated hypotheses using cost functions for abduced events and object tracks.

Weights for events and motion.

```
%% weights for calculating cost of tracks

weight(noise, 500).
weight(missing_det, 10).
weight(missing_till_end, 5).
weight(missing_from_start, 5).
weight(stands, 30).
weight(exists, 30).
weight(occludes, 1).
weight(no_exp_start, 1000).
weight(no_exp_end, 1000).
weight(missing_det_vel, 10).
weight(missing_vel, 10).
```

Costs for events based on event type and length.

```
%% Optimization of costs for events and length of events

#minimize {} (T_end-T_start) * ALPHA_1, TR : noise(TR),
  ends(TR, T_end), starts(TR, T_start),
  weight(noise, ALPHA_1)).

#minimize {} (T2_start-T1_end) * ALPHA_1, TR1, TR2 :
  missing_det(TR1,TR2,T1,T2),
  ends(TR1, T1_end), starts(TR2, T2_start),
  weight(missing_det, ALPHA_1)).

#minimize {(T, T2_end) T_end | ALPHA_1, TR : missing_till_end(TR, T),
  ends(TR, T_end), shot(_, T_shot_end),
  weight(missing_till_end, ALPHA_1)).

#minimize {} (T_start-T_shot_start) * ALPHA_1, TR :
  missing_from_start(TR, T),
  starts(TR, T_start), shot(T_shot_start, _),
  weight(missing_from_start, ALPHA_1)).
```
% optimization continue

#minimize {ALPHA_1, TR : no_exp_start{TR,_,},
weight(no_exp_start, ALPHA_1)}.

#minimize {ALPHA_1, TR : no_exp_end{TR,_,},
weight(no_exp_end, ALPHA_1)}.

#minimize {ALPHA_1, TR : enters(_ , TR,_,),
weight(enters, ALPHA_1)}.

#minimize {ALPHA_1, TR : exits(_, TR,_,),
weight(exits, ALPHA_1)}.

#minimize {{T3_start-T2_end}+ALPHA_1, TR1, TR2, TR3 : occludes(TR1,TR2,TR3,T2_end,T3_start),
ends(TR2, T2_end), starts(TR3, T3_start),
weight(occludes, ALPHA_1)}.

Cost for abduced motion tracks.

% Optimization of cost for motion, i.e., constant velocity

#minimize {((X_vel_prev-2*X_vel_during+X_vel_next)**2+
(Y_vel_prev-2*Y_vel_during+Y_vel_next)**2)*ALPHA_1, TR1, TR2 :
missing_det(TR1,TR2,T1,T2),
track(TR1, Box_T1, T1),
box(Box_T1, X_e, Y_e, _ , _),
track(TR1, Box_T1_prev, T1-1),
box(Box_T1_prev, X_e_prev, Y_e_prev, _ , _),
track(TR2, Box_T2, T2),
box(Box_T2, X_s, Y_s, _ , _),
track(TR2, Box_T2_next, T2+1),
box(Box_T2_next, X_s_next, Y_s_next, _ , _),
X_vel_during = (X_e - X_s) / (T2-T1),
Y_vel_during = (Y_e - Y_s) / (T2-T1),
X_vel_prev = (X_e_prev - X_e) / 1,
Y_vel_prev = (Y_e_prev - Y_e) / 1,
X_vel_next = (X_s - X_s_next) / 1,
Y_vel_next = (Y_s - Y_s_next) / 1,
weight(missing_det_vel, ALPHA_1)}.

#minimize {((X_vel_prev-2*X_vel_during+X_vel_next)**2+
(Y_vel_prev-2*Y_vel_during+Y_vel_next)**2)*ALPHA_1, TR1, TR2 :
occludes(_,TR1,TR2,T1,T2),
track(TR1, Box_T1, T1),
box(Box_T1, X_e, Y_e, _ , _),
track(TR1, Box_T1_prev, T1-1),
box(Box_T1_prev, X_prev, Y_prev, _ , _),
track(TR2, Box_T2, T2),
box(Box_T2, X_s, Y_s, _ , _),
track(TR2, Box_T2_next, T2+1),
box(Box_T2_next, X_s_next, Y_s_next, _ , _),
X_vel_during = (X_e - X_s) / (T2-T1),
Y_vel_during = (Y_e - Y_s) / (T2-T1),
X_vel_prev = (X_prev - X_e) / 1,
Y_vel_prev = (Y_prev - Y_e) / 1,
X_vel_next = (X_s - X_s_next) / 1,
Y_vel_next = (Y_s - Y_s_next) / 1,
weight(occludes_vel, ALPHA_1)}.
Figure 5: Sample Scene from the MOT dataset: corrected tracks in challenging setting (noisy detections and many targets). First Row: original video. Second Row: Object Tracks obtained from Tracking. Third Row: Corrected People Tracks.

Figure 6: Sample Scene from the movie “The Bad Sleep Well” by Akira Kurusawa (1960): abducted high-level event passing behind and corrected people tracks. First Row: original video. Second Row: Object Tracks obtained from Tracking. Third Row: Corrected People Tracks.

Figure 7: Sample Scene from the movie “Drive” by Nicolas Winding Refn (2011): abducted missing detections and corrected people tracks. First Row: original video. Second Row: Object Tracks obtained from Tracking. Third Row: Corrected People Tracks.