Textual Explanations for Self-Driving Vehicles

Jinkyu Kim[1] [0000−0001−6520−2074], Anna Rohrbach[1,2] [0000−0003−1161−6006],
Trevor Darrell[1] [0000−0001−5453−8533], John Canny[1] [0000−0002−7161−7927], and
Zeynep Akata[2,3] [0000−0002−1432−7747]

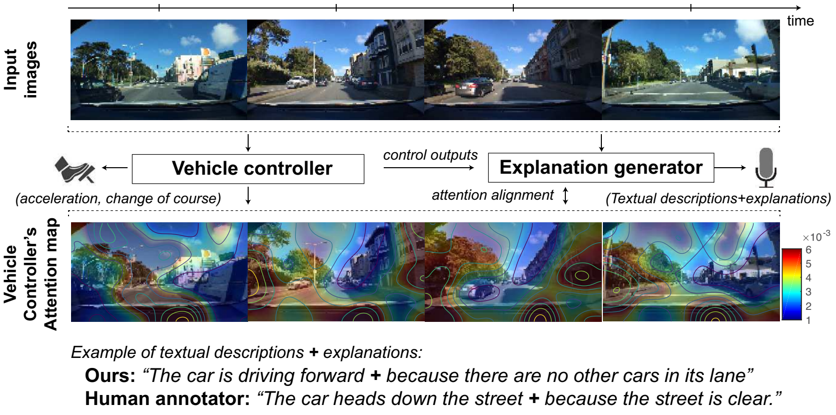
[1] EECS, University of California, Berkeley CA 94720, USA
[2] MPI for Informatics, Saarland Informatics Campus, 66123 Saarbrücken, Germany
[3] AMLab, University of Amsterdam, 1098 XH Amsterdam, Netherlands
Correspondence: z.akata@uva.nl

Abstract. Deep neural perception and control networks have become key components of self-driving vehicles. User acceptance is likely to benefit from easy-to-interpret textual explanations which allow end-users to understand what triggered a particular behavior. Explanations may be triggered by the neural controller, namely *introspective explanations*, or informed by the neural controller's output, namely *rationalizations*. We propose a new approach to introspective explanations which consists of two parts. First, we use a visual (spatial) attention model to train a convolutional network end-to-end from images to the vehicle control commands, *i.e.*, acceleration and change of course. The controller's attention identifies image regions that potentially influence the network's output. Second, we use an attention-based video-to-text model to produce textual explanations of model actions. The attention maps of controller and explanation model are aligned so that explanations are grounded in the parts of the scene that mattered to the controller. We explore two approaches to attention alignment, strong- and weak-alignment. Finally, we explore a version of our model that generates rationalizations, and compare with introspective explanations on the same video segments. We evaluate these models on a novel driving dataset with ground-truth human explanations, the Berkeley DeepDrive eXplanation (BDD-X) dataset. Code is available at `https://github.com/JinkyuKimUCB/explainable-deep-driving`

Keywords: Explainable Deep Driving · BDD-X dataset

1 Introduction

Deep neural networks are an effective tool [3,26] to learn vehicle controllers for self-driving cars in an end-to-end manner. Despite their effectiveness as function estimators, DNNs are typically cryptic black-boxes. There are no explainable states or labels in such a network, and representations are fully distributed as sets of activations. Explainable models that make deep models more transparent are important for a number of reasons: (i) user acceptance – self-driving vehicles are a radical technology for users to accept, and require a very high level of trust, (ii) understanding and extrapolation of vehicle behavior – users ideally should be able to anticipate what the vehicle will do in most situations, (iii) effective communication – they help user communicate preferences to the vehicle and vice versa.


time
Input images
Vehicle controller
control outputs
Explanation generator
(acceleration, change of course)
attention alignment
(Textual descriptions+explanations)
Vehicle Controller's Attention map
×10^-3
6
5
4
3
2
1

Example of textual descriptions **+** *explanations:*
Ours: *"The car is driving forward* **+** *because there are no other cars in its lane"*
Human annotator: *"The car heads down the street* **+** *because the street is clear."*

Fig. 1: Our model predicts vehicles control commands, *i.e.*, an acceleration and a change of course, at each timestep, while an explanation model generates a natural language explanation of the rationales, *e.g.*, "The car is driving forward because there are no other cars in its lane", and a visual explanation in the form of attention – attended regions directly influence the textual explanation generation process.

Explanations can be either *rationalizations* – explanations that justify the system's behavior in a post-hoc manner, or *introspective explanations* – explanations that are based on the system's internal state. Introspective explanations represent *causal* relationships between the system's input and its behavior, and address all the goals above. Rationalizations can address acceptance, (i) above, but are less helpful with (ii) understanding the causal behavior of the model or (iii) communication which is grounded in the vehicle's internal state (known as theory of mind in human communication).

One way of generating introspective explanations is via visual attention [27,11]. Visual attention filters out non-salient image regions, and image areas inside the attended region have potential causal effect on the output (those outside cannot). As shown in [11], additional salience filtering can be applied so that the attention map shows only regions that causally affect the output. Visual attention constrains the reasons for the controllers actions but does not *e.g.*, tie specific actions to specific input regions *e.g.*, "the vehicle slowed down because the light controlling the intersection is red". It is also likely to be less convenient for passengers to replay the attention map vs. a (typically on-demand) speech presentation of a textual explanation.

In this work, we focus on generating textual descriptions and explanations, such as the pair: "vehicle slows down" and "because it is approaching an intersection and the light is red" as in Figure 1. Natural language has an advantage of being inherently understandable and does not require familiarity with the design of an intelligent system in order to provide useful information. In order to train such a model, we collect explanations from human annotators. Our explanation dataset is built on top of another large-scale driving dataset [26] collected from dashboard cameras in human driven vehicles. Annotators view the video dataset, compose descriptions of the vehicle's activity and explanations for the actions that the vehicle driver performed.

Obtaining training data for vehicle explanations is by itself a significant challenge. The ground truth explanations are in fact often rationalizations (generated by an observer rather than the driver), and there are additional challenges with acquiring driver data. But even more than that, it is currently impossible to obtain human explanations of *what the vehicle controller was thinking*, *i.e.*, a real ground truth. Nevertheless our experiments show that using *attention alignment* between controller and explanation models generally improves the quality of explanations, *i.e.*, generates explanations which better match the human rationalizations of the driving videos.

Our contributions are as follows. (1) We propose an introspective textual explanation model for self-driving cars to provide easy-to-interpret explanations for the behavior of a deep vehicle control network. (2) We integrate our explanation generator with the vehicle controller by aligning their attentions to ground the explanation, and compare two approaches: attention-aligned explanations and non-aligned rationalizations. (3) We generated a large-scale Berkeley DeepDrive eXplanation (BDD-X) dataset with over 6,984 video clips annotated with driving descriptions, *e.g.*, "The car slows down" and explanations, *e.g.*, "because it is about to merge with the busy highway". Our dataset provides a new test-bed for measuring progress towards developing explainable models for self-driving cars.

2 Related Work

In this section, we review existing work on end-to-end learning for self-driving cars as well as work on visual explanation and justification.

End-to-End Learning for Self-Driving Cars: Most of vehicle controllers for self-driving cars can be divided in two types of approaches [5]: (1) a mediated perception-based approach and (2) an end-to-end learning approach. The mediated perception-based approach depends on recognizing human-designated features, such as lane markings, traffic lights, pedestrians or cars, which generally require demanding parameter tuning for a balanced performance [19]. Notable examples include [23], [4], and [16].

As for the end-to-end approaches, recent works [3,26] suggest that neural networks can be successfully applied to self-driving cars in an end-to-end manner. Most of these approaches use behavioral cloning that learns a driving policy as a supervised learning problem over observation-action pairs from human driving demonstrations. Among these, [3] present a deep neural vehicle controller network that directly maps a stream of dashcam images to steering controls, while [26] use a deep neural network that takes input raw pixels and prior vehicle states and predict vehicle's future motion. Despite their potential, the effectiveness of these approaches is limited by their inability to explain the rationale for the system's decisions, which makes their behavior opaque and uninterpretable. In this work, we propose an end-to-end trainable system for self driving cars that is able to justify its predictions visually via attention maps and textually via natural language.

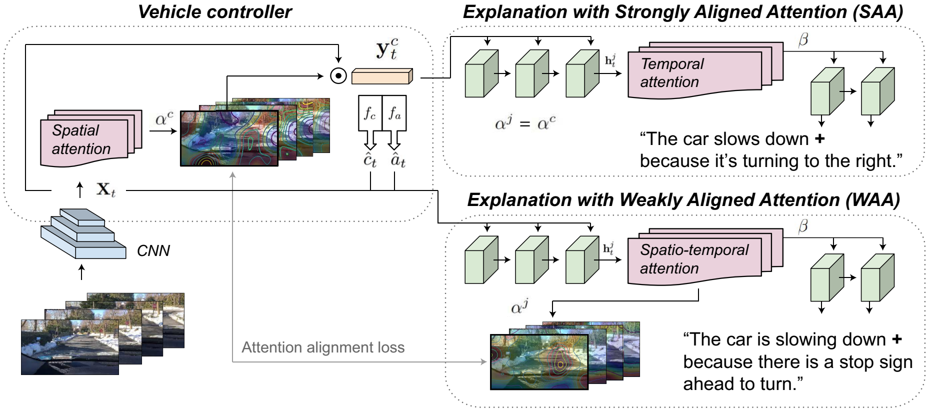

Vehicle controller
Explanation with Strongly Aligned Attention (SAA)
$\mathbf{y}_t^c$
$\odot$
α^c
Spatial attention
f_c
f_a
$\hat{c}_t$
$\hat{a}_t$
$\mathbf{X}_t$
CNN
$\mathbf{h}_t^j$
Temporal attention
β
$\alpha^j = \alpha^c$
"The car slows down + because it's turning to the right."
Explanation with Weakly Aligned Attention (WAA)
$\mathbf{h}_t^j$
Spatio-temporal attention
β
α^j
Attention alignment loss
"The car is slowing down + because there is a stop sign ahead to turn."

Fig. 2: Vehicle controller generates spatial attention maps α^c for each frame, predicts acceleration and change of course $(\hat{c}_t, \hat{a}_t)$ that condition the explanation. Explanation generator predicts temporal attention across frames (β) and a spatial attention in each frame (α^j). SAA uses α^c, WAA enforces a loss between α^j and α^c.

Visual and Textual Explanations: The importance of explanations for an end-user has been studied from the psychological perspective [17,18], showing that humans use explanations as a guide for learning and understanding by building inferences and seeking propositions or judgments that enrich their prior knowledge. They usually seek for explanations to fill the requested gap depending on prior knowledge and goal in question.

In support of this trend, recently explainability has been growing as a field in computer vision and machine learning. Especially, there is a growing interest in introspective deep neural networks. [28] use deconvolution to visualize inner-layer activations of convolutional networks. [14] propose automatically-generated captions for textual explanations of images. [2] develop a richer notion of contribution of a pixel to the output. However, a difficulty with deconvolution-style approaches is the lack of formal measures of how the network output is affected by spatially-extended features (rather than pixels). Exceptions to this rule are attention-based approaches. [11] propose attention-based approach with causal filtering that removes spurious attention blobs. However, it is also important to be able to justify the decisions that were made and explain why they are reasonable in a human understandable manner, *i.e.*, a natural language. For an image classification problem, [7,8] used an LSTM [9] caption generation model that generates textual justifications for a CNN model. [21] combine attention-based model and a textual justification system to produce an interpretable model. To our knowledge, ours is the first attempt to justify the decisions of a real-time deep controller through a combination of attention and natural language explanations on a stream of images.

3 Explainable Driving Model

In this paper, we propose a driving model that explains how a driving decision was made both (i) by visualizing image regions where the decision maker attends to and (ii) by generating a textual description and explanation of what has triggered a particular driving decision, *e.g.*, "the car continues (description) because traffic flows freely (explanation)". As we summarize in Figure 2, our model involves two parts: (1) a *Vehicle controller*, which is trained to learn human-demonstrated vehicle control commands, *e.g.*, an acceleration and a change of course; our controller uses a visual (spatial) attention mechanism that identifies potentially influential image regions for the network's output; (2) a *Textual explanation generator*, which generates textual descriptions and explanations controller behavior. The key to the approach is to align the attention maps.

Preprocessing. Our model is trained to predict two vehicle control commands, *i.e.*, an acceleration and a change of course. At each time t, an acceleration, a_t, is measured by taking the derivative of speed measurements, and a change of course, c_t, is computed by taking a difference between a current vehicle's course and a smoothed value by using simple exponential smoothing method [10]. We provide details in supplemental material. To reduce computational burden, we down-sample to 10Hz and reduce the input dimensionality by resizing raw images to a $90\times160\times3$ image with nearest-neighbor scaling algorithm. Each image is then normalized by subtracting the mean from the raw input pixels and dividing by its standard deviation. This preprocessing is applied to the latest 4 frames, which are then stacked to produce the final input to the neural network.

Convolutional Feature Encoder. We use a convolutional neural network to encode the visual information into a set of visual feature vectors at time t, *i.e.*, convolutional feature cube $\mathbf{X}_t = \{\mathbf{x}_{t,1}, \mathbf{x}_{t,2}, \ldots, \mathbf{x}_{t,l}\}$ where $\mathbf{x}_{t,i} \in \mathcal{R}^d$ for $i \in \{1, 2, \ldots, l\}$ and l is the number of different spatial regions of the given input. Each feature vector contains a high-level description of objects present in a certain input region. This allows us to focus selectively on different regions of the given image by choosing a subset of these feature vectors. We use a five-layered convolutional network as in [3,11] and omit max-pooling layers to prevent spatial information loss [15]. The output is a three-dimensional feature cube $\mathbf{X}_t$ and the feature block has the size $w\times h\times d$ at each time t.

3.1 Vehicle Controller

Our vehicle controller is trained in an end-to-end manner. Given a stream of dashcam images and the vehicle's (current) sensor measurements, *e.g.*, speed, the controller predicts the acceleration and the change of course at each timestep. We utilize a deterministic soft attention mechanism that is trainable by standard back-propagation methods. The soft attention mechanism applies attention weights multiplicatively to the features and additively pools the results through the maps π. Our model feeds the context vectors $\mathbf{y}_t^c$ produced by the controller map π^c to the controller LSTM:

$$\mathbf{y}_t^c = \pi^c(\{\alpha_{t,i}^c\}, \{\mathbf{x}_{t,i}\}) = \sum_{i=1}^{l} \alpha_{t,i}^c \mathbf{x}_{t,i} \tag{1}$$

where $i = \{1, 2, \ldots, l\}$. $\alpha^c_{t,i}$ is an attention weight map output by a spatial softmax and satisfies $\sum_i \alpha^c_{t,i} = 1$. These attention weights can be interpreted as the probability over l convolutional feature vectors. A location with a high attention weight is salient for the task (driving). The attention model $f^c_{\text{attn}}(\mathbf{X}_t, \mathbf{h}^c_{t-1})$ is conditioned on the previous LSTM state $\mathbf{h}^c_{t-1}$, and the current feature vectors $\mathbf{X}_t$. It comprises a fully-connected layer and a spatial softmax to yield normalized $\{\alpha^c_{t,i}\}$.

The outputs of the vehicle controller are the vehicle's acceleration $\hat{a}_t$ and the change of course $\hat{c}_t$. To this end, we use additional multi-layer fully-connected blocks with ReLU non-linearities, denoted by $f_{\text{a}}(\mathbf{y}^c_t, \mathbf{h}^c_t)$ and $f_{\text{c}}(\mathbf{y}^c_t, \mathbf{h}^c_t)$. We also add the entropy H of the attention weight to the objective function:

$$\mathcal{L}_{\text{c}} = \sum_t \left((a_t - \hat{a}_t)^2 + (c_t - \hat{c}_t)^2 + \lambda_{\text{c}} H(\alpha^c_t)\right) \tag{2}$$

The entropy is computed on the attention map as though it were a probability distribution. Minimizing loss corresponds to minimizing entropy. Low entropy attention maps are sparse and emphasize relatively few regions. We use a hyperparameter λ_{c} to control the strength of the entropy regularization term.

3.2 Attention Alignments

The controller attention map provides input regions that the network attends to, and these regions have a direct influence on the network's output. Thus, to yield "introspective" explanation, we argue that the agent must attend to those areas. For example, if a vehicle controller predicts "acceleration" by detecting a green traffic light, the textual justification must mention this evidence, *e.g.*, "because the light has turned green". Here, we explain two approaches to align the vehicle controller and the textual justifier such that they look at the same input regions.

Strongly Aligned Attention (SAA): A consecutive set of spatially attended input regions, each of which is encoded as a context vector $\mathbf{y}^c_t$ by the vehicle controller, can be directly used to generate a textual explanation (see Figure 2, right-top). Thus, models share a single layer of an attention. As we detail in Section 3.3, our explanation module uses *temporal* attention with weights β to the controller context vectors $\{\mathbf{y}^j_t, t = 1, \ldots\}$ directly, and thus allows flexibility in output tokens relative to input samples.

Weakly Aligned Attention (WAA): Instead of directly using vehicle controller's attention, an explanation generator can have its own spatial attention network (see Figure 2, right-bottom). A loss, *i.e.*, the Kullback-Leibler divergence (D_{KL}), between the two attention maps makes the explanation generator refer to the salient objects:

$$\mathcal{L}_a = \lambda_a \sum_t D_{\text{KL}}(\alpha^c_t || \alpha^j_t) = \lambda_a \sum_t \sum_{i=1}^{l} \alpha^c_{t,i}(\log \alpha^c_{t,i} - \log \alpha^j_{t,i}) \tag{3}$$

where α^c and α^j are the attention maps generated by the vehicle controller and the explanation generator model, respectively. We use a hyperparameter λ_a to control the strength of the regularization term.

3.3 Textual Explanation Generator

Our textual explanation generator takes sequence of video frames of variable length and generates a variable-length description/explanation. Descriptions and explanations are typically part of the same sentence in the training data but are annotated with a separator. In training and testing we use a synthetic separator token `<sep>` between description and explanation, but treat them as a single sequence. The explanation LSTM predicts the description/explanation sequence and outputs per-word softmax probabilities.

The source of context vectors for the description generator depends on the type of alignment between attention maps. For weakly aligned attention or rationalizations, the explanation generator creates its own spatial attention map α^j at each time step t. This map includes a loss against the controller attention map for weakly-aligned attention, but has no such loss when generating rationalizations. The attention map α^j is applied to the CNN output yielding context vectors $\mathbf{y}_t^j$.

Our textual explanation generator explains the rationale behind the driving model, and thus we argue that a justifier needs the outputs from the vehicle motion predictor as an input. We concatenate a tuple $(\hat{a}_t, \hat{c}_t)$ with a spatially-attended context vector $\mathbf{y}_t^j$ and $\mathbf{y}_t^c$ respectively for weakly and strongly aligned attention approaches. This concatenated vector is then used to update the LSTM for a textual explanation generation.

The explanation module applies *temporal* attention with weights β to either the controller context vectors directly $\{\mathbf{y}_t^c, t = 1, \ldots\}$ (strong alignment), or to the explanation vectors $\{\mathbf{y}_t^j, t = 1, \ldots\}$ (weak alignment or rationalization). Such input sequence attention is common in sequence-to-sequence models and allows flexibility in output tokens relative to input samples [1]. The result of temporal attention application is (dropping the c or j superscripts on $\mathbf{y}$):

$$\mathbf{z}_k = \pi(\{\beta_{k,t}\}, \{\mathbf{y}_t\}) = \sum_{t=1}^{T} \beta_{k,t}\mathbf{y}_t \tag{4}$$

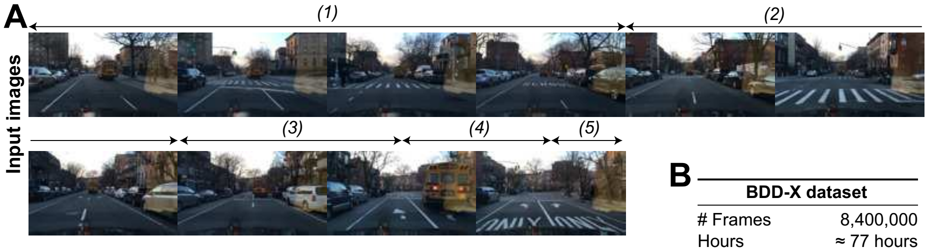
where $\sum_t \beta_{k,t} = 1$. The weight $\beta_{k,t}$ at each time k (for sentence generation) is computed by an attention model $f^e_{\text{attn}}(\{\mathbf{y}_t\}, \mathbf{h}^e_{k-1})$, which is similar to the spatial attention as we explained in previous section (see supplemental material for details).

To summarize, we minimize the following negative log-likelihood (for training our justifier) as well as vehicle control estimation loss $\mathcal{L}_c$ and attention alignment loss $\mathcal{L}_a$:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_a - \sum_k \log p(\mathbf{o}_k|\mathbf{o}_{k-1}, h^e_k, \mathbf{z}_k) \tag{5}$$

4 Berkeley DeepDrive eXplanation Dataset (BDD-X)

In order to effectively generate and evaluate textual driving rationales we have collected textual justifications for a subset of the Berkeley Deep Drive (BDD) dataset [26]. This dataset contains videos, approximately 40 seconds in length, captured by a dashcam mounted behind the front mirror of the vehicle. Videos are mostly captured during urban driving in various weather conditions, featuring day and nighttime. The dataset



Action descriptions:	Action explanations:
(1) The car is driving	***as** there is nothing to impede it.*
(2) The car is moving into the right lane	***because** it is safe to do so.*
(3) The car moves back into the left lane	***because** the school bus in front of it is stopping.*
(4) The car drives in the left lane	***in order to** pass the school bus.*
(5) The car moves into the right lane	***since** it has now passed the school bus and it is taking the right fork.*

B

BDD-X dataset	
# Frames	8,400,000
Hours	≈ 77 hours
Condition	Urban
Lighting	Day/Night
# Annotations	26,228
Avg. # actions / videos	3.8
# Videos	6,984
# Training	5,588
# Validation / Testing	698

Fig. 3: (A) Examples of input frames and corresponding human-annotated action description and justification of how a driving decision was made. For visualization, we sample frames at every two seconds. (B) BDD-X dataset details. Over 77 hours of driving with time-stamped human annotations for action descriptions and justifications.

also includes driving on other road types, such as residential roads (with and without lane markings), and contains all the typical driver's activities such as staying in a lane, turning, switching lanes, etc. Alongside the video data, the dataset provides a set of time-stamped sensor measurements, such as vehicle's velocity, course, and GPS location. For sensor logs unsynchronized with the time-stamps of video data, we use the estimates of the interpolated measurements.

In order to increase trust and reliability, the machine learning system underlying self driving cars should be able to explain why at a certain time they make certain decisions. Moreover, a car that justifies its decision through natural language would also be user friendly. Hence, we populate a subset of the BDD dataset with action description and justification for all the driving events along with their timestamps. We provide examples from our *Berkeley Deep Drive eXplanation (BDD-X)* dataset in Figure 3 (A).

Annotation. We provide a driving video and ask a human annotator in Amazon Mechanical Turk to imagine herself being a driving instructor. Note that we specifically select human annotators who are familiar with US driving rules. The annotator has to describe *what* the driver is doing (especially when the behavior changes) and *why*, from a point of view of a driving instructor. Each described action has to be accompanied with a start and end time-stamp. The annotator may stop the video, forward and backward through it while searching for the activities that are interesting and justifiable.

To ensure that the annotators provide us the driving rationales as well as descriptions, we require that they separately enter the *action description* and the *action justification*: *e.g.*, *"The car is moving into the left lane"* and *"because the school bus in front of it is stopping."*. In our preliminary annotation studies, we found that giving separate annotation boxes is helpful for the annotator to understand the task and perform better.

Dataset Statistics. Our dataset (see Figure 3 (B)) is composed of over 77 hours of driving within 6,984 videos. The videos are taken in diverse driving conditions, *e.g.*, day/night, highway/city/countryside, summer/winter etc. On an average of 40 seconds, each video contains around 3-4 actions, *e.g.*, speeding up, slowing down, turning right etc., all of which are annotated with a description and an explanation. Our dataset contains over $26K$ activities in over $8.4M$ frames. We introduce a training, a validation and a test set, containing 5,588, 698 and 698 videos, respectively.

Inter-human agreement. Although we cannot have access to the internal thought process of the drivers, one can infer the reason behind their actions using the visual evidence of the scene. Besides, it would be challenging to setup the data collection process which enables drivers to report justifications for all their actions, if at all possible. We ensure the high quality of the collected annotations by relying on a pool of qualified workers (*i.e.*, they pass a qualification test) and selective manual inspection.

Further, we measure the inter-human agreement on a subset of 998 training videos, each of which has been annotated by two different workers. Our analysis is as follows. In 72% of videos the number of annotated intervals differs by less than 3. The average temporal IoU across annotators is 0.63 ($SD = 0.21$). When $IoU > 0.5$ the CIDEr score across action descriptions is 142.60, across action justifications it is 97.49 (random choice: 39.40/28.39, respectively). When $IoU > 0.5$ and action descriptions from two annotators are identical (165 clips[1]) the CIDEr score across justifications is 200.72, while a strong baseline, selecting a justification from a different video with the same action description, results in CIDEr score 136.72. These results show an agreement among annotators and relevance of collected action descriptions and justifications.

Coverage of justifications. BDD-X dataset has over 26k annotations (77 hours) collected from a substantial random subset of large-scale crowd-sourced driving video dataset, which consists of all the typical drivers activities during urban driving. The vocabulary of training action descriptions and justifications is 906 and 1,668 words respectively, suggesting that justifications are more diverse than descriptions. Some of the common actions are (frequency decreasing): moving forward, stopping, accelerating, slowing, turning, merging, veering, pulling [in]. Justifications cover most of the relevant concepts: traffic signs/lights, cars, lanes, crosswalks, passing, parking, pedestrians, waiting, blocking, safety etc.

5 Results and Discussion

Here, we first provide our training and evaluation details, then make a quantitative and qualitative analysis of our vehicle controller and our textual justifier.

Training and Evaluation Details. As the convolutional feature encoder, we use 5-layer CNN [3] that produces a 12×20×64-dimensional convolutional feature cube from the last layer. The controller following the CNN has 5 fully connected layers (*i.e.*, #hidden dims: 1164, 100, 50, 10, respectively), which predict the acceleration and the change of

[1] The number of video intervals (not full videos), where the provided action descriptions (not explanations) are identical (common actions *e.g.*, “the car slows down”).

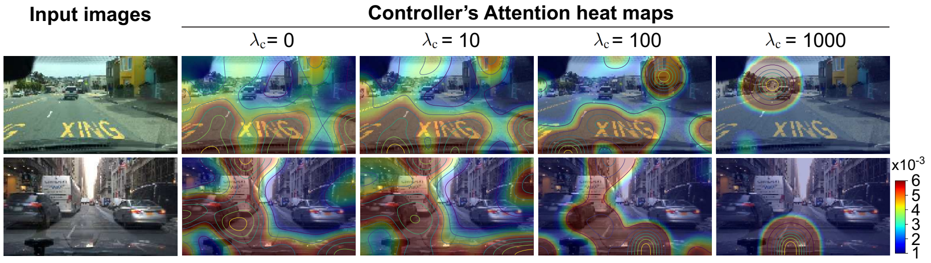


Fig. 4: Vehicle controllers attention maps in terms of four different entropy regularization coefficient λ_c={0,10,100,1000}. Red parts indicate where the model pays more attention. Higher value of λ_c makes the attention maps sparser. We observe that sparser attention maps improves the performance of generating textual explanations, while control performance is slightly degraded.

course, and is trained end-to-end from scratch. Using other more expressive networks may give a performance boost over our base CNN configuration, but these explorations are out of our scope. Given the obtained convolutional feature cube, we first train our vehicle controller, and then the explanation generator (single layer LSTM unless stated otherwise) by freezing the control network. For training, we use Adam optimizer [12] and dropout [22] of 0.5 at hidden state connections and Xavier initialization [6]. The standard dataset is split as 80% (5,588 videos) as the training set, 10% (698 videos) as the test, and 10% (698 videos) as the validation set. Our model takes less than a day to train on a single NVIDIA Titan X GPU.

For evaluating the vehicle controller we use the mean absolute error (lower is better) and the distance correlation (higher is better) and for the justifier we use BLEU [20], METEOR [13], and CIDEr-D [24], as well as human evaluation. The former metrics are widely used for the evaluation of video and image captioning models automatically against ground truth.

5.1 Evaluating Vehicle Controller

We start by quantitatively comparing variants of our vehicle controller and the state of the art, which include variants of the work by Bojarski *et al.* [3] and Kim *et al.* [11] in Table 1. Note that these works differ from ours in that their output is the curvature of driving, while our model estimates continuous acceleration and the change of course values. Thus, their models have a single output, while ours estimate both control commands. In this experiment, we replaced their output layer with ours. For a fair comparison, we use an identical CNN for all models.

In this experiment, each model estimates vehicle's acceleration and the change of course. Our vehicle controller predicts acceleration and the change of course, which generally requires prior knowledge of vehicle's current state, *i.e.*, speed and course, and navigational inputs, especially in urban driving. We observe that the use of the latest

Model	λ_c	Mean of absolute error (MAE)		Mean of distance correlation	
		Acceleration (m/s^2)	Course (degree)	Acceleration (m/s^2)	Course (degree)
CNN+FC [3]†	-	6.92 [7.50]	12.1 [19.7]	0.17 [0.15]	0.16 [0.14]
CNN+FC [3]+P	-	6.09 [7.73]	6.74 [14.9]	0.21 [0.18]	0.39 [0.33]
CNN+LSTM+Attention [11]†	-	6.87 [7.44]	10.2 [18.4]	0.19 [0.16]	0.22 [0.18]
CNN+LSTM+Attention+P (Ours)	1000	5.02 [6.32]	6.94 [15.4]	0.65 [0.25]	0.43 [0.33]
CNN+LSTM+Attention+P (Ours)	100	2.68 [3.73]	6.17 [14.7]	0.78 [0.28]	0.43 [0.34]
CNN+LSTM+Attention+P (Ours)	10	2.33 [3.38]	6.10 [14.7]	0.81 [0.27]	0.46 [0.35]
CNN+LSTM+Attention+P (Ours)	0	**2.29 [3.33]**	**6.06 [14.7]**	**0.82 [0.26]**	**0.47 [0.35]**

Table 1: Comparing variants of our vehicle controller with different values of the entropy regularization coefficient λ_c=$\{0, 10, 100, 1000\}$ and the state-of-the-art. High value of λ_c produces low entropy attention maps that are sparse and emphasize relatively few regions. †: Models use a single image frame as an input. The standard deviation is in braces. *Abbreviation:* FC (fully connected layer), P (prior inputs)

four consecutive frames and prior inputs (*i.e.*, vehicle's motion measurement and navigational information) improves the control prediction accuracy (see 3rd vs. 7th row), while the use of visual attention also provides improvements (see 1st vs. 3rd row). Specifically, our model without the entropy regularization term (last row) performs the best compared to CNN based approaches [3] and [11]. The improvement is especially pronounced for acceleration estimation.

In Figure 4 we compare input images (first column) and corresponding attention maps for different entropy regularization coefficients λ_c=$\{0, 10, 100, 1000\}$. Red is high attention, blue is low. As we see, higher λ_c lead to sparser maps. For better visualization, an attention map is overlaid by its contour lines and an input image.

Quantitatively, the controller performance (error and correlation) slightly degrade as λ_c increases and the attention maps become more sparse (see bottom four rows in Table 1). So there is some tension between sparse maps (which are more interpretable), and controller performance. An alternative to regularization, [11] use causal filtering over the controller's attention maps and achieve about 60% reduction in "hot" attention pixels. Causal filtering is desirable for the present work not only to improve sparseness but because after causal filtering, "hot" regions necessarily *do* have a causal effect on controller behavior, whereas unfiltered attention regions may not. We will explore it in future work.

5.2 Evaluating Textual Explanations

In this section, we evaluate textual explanations against the ground truth explanation using automatic evaluation measures, and also provide human evaluation followed by a qualitative analysis.

Automatic Evaluation. For state-of-the-art comparison, we implement the S2VT [25] and its variants. Note that in our implementation S2VT uses our CNN and does not

Type	Model	Control inputs	λ_a	λ_c	Explanations (*e.g.*, *"because the light is red"*)			Descriptions (*e.g.*, *"the car stops"*)		
					BLEU-4	METEOR	CIDEr-D	BLEU-4	METEOR	CIDEr-D
	S2VT [25]	N	-	-	6.332	11.19	53.35	30.21	27.53	179.8
	S2VT [25]+SA	N	-	-	5.668	10.96	51.37	28.94	26.91	171.3
	S2VT [25]+SA+TA	N	-	-	5.847	10.91	52.74	27.11	26.41	157.0
Rationalization	Ours (no constraints)	Y	0	0	6.515	12.04	61.99	31.01	28.64	205.0
Introspective explanation	Ours (with SAA)	Y	-	0	6.998	12.08	62.24	**32.44**	29.13	213.6
	Ours (with SAA)	Y	-	10	6.760	12.23	63.36	29.99	28.26	203.6
	Ours (with SAA)	Y	-	100	7.074	12.23	66.09	31.84	29.11	214.8
	Ours (with WAA)	Y	10	0	6.967	12.14	64.19	32.24	29.00	**219.7**
	Ours (with WAA)	Y	10	10	6.951	**12.34**	68.56	30.40	28.57	206.6
	Ours (with WAA)	Y	10	100	**7.281**	12.24	**69.52**	32.34	**29.22**	215.8

Table 2: Comparing generated and ground truth (columns 6-8) descriptions (*e.g.*, "the car stops") and explanations (*e.g.*, "because the light is red"). We implement S2VT [25] and variants with spatial attention (SA) and temporal attention (TA) as a baseline. We tested two different attention alignment approaches, *i.e.*, WAA (weakly aligned attention) and SAA (strongly aligned attention), with different combinations of two regularization coefficients: λ_a=$\{0, 10\}$ for the attention alignment and λ_c=$\{0, 10, 100\}$ for the vehicle controller. Rationalization baseline relies on our model (WAA approach) but has no attention alignment. Note that we report all values as a percentage.

use optical flow features. In Table 2, we report a summary of our experiment validating the quantitative effectiveness of our approach. Rows 5-10 show that best explanation results are generally obtained with weakly-aligned attention. Comparing with row 4, the introspective models all gave higher scores than the rationalization model for explanation generation. Description scores are more mixed, but most of the introspective model scores are higher. As we will see in the next section, our rationalization model focuses on visual saliencies, which is sometimes different from what controller actually "looks at". For example, in Figure 5 (5th example), our controller sees the front vehicle and our introspective models generate explanations such as "because the car in front is moving slowly", while our rationalization model does not see the front vehicle and generates "because it's turning to the right".

As our training data are human observer annotations of driving videos, and they are not the explanations of drivers, they are post-hoc rationalizations. However, based on the visual evidence, (*e.g.*, the existence of a turn right sign explains why the driver has turned right even if we do not have access to the exact thought process of the driver), they reflect typical causes of human driver behavior. The data suggest that grounding the explanations in controller internal state helps produce explanations that better align with human third-party explanations. Biasing the explanations toward controller state (which the WAA and SAA models do) improves their plausibility from a human perspective, which is a good sign. We further analyze human preference in the evaluation below.

Type	Model	Control inputs	λ_a	λ_c	Correctness rate	
					Explanations	Descriptions
Rationalization	Ours (no constraints)	Y	0	0	64.0%	92.8%
Introspective explanation	Ours (with SAA)	Y	-	100	62.4%	90.8%
	Ours (with WAA)	Y	10	100	**66.0%**	**93.5%**

Table 3: Human evaluation of the generated action descriptions and explanations for randomly chosen 250 video intervals. We measure the success rate where at least 2 human judges rate the generated description or explanation with a score 1 (correct and specific/detailed) or 2 (correct).

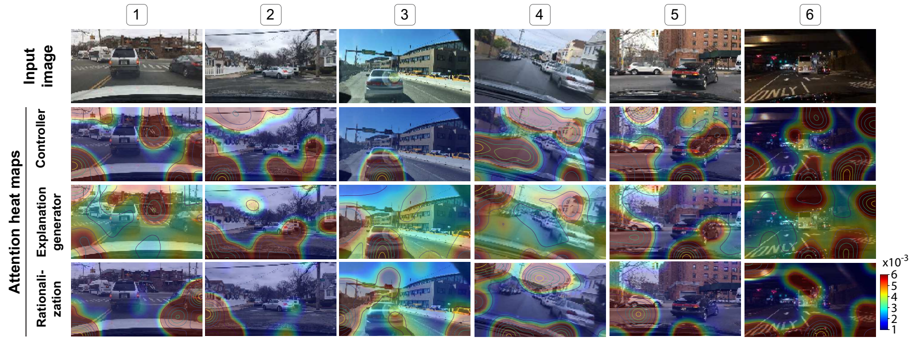
Human Evaluation. In our first human evaluation experiment the human judges are only shown the *descriptions*, while in the second experiment they only see the *explanations* (e.g. "*The car ... because* $< explanation >$"), to exclude the effect of explanations/descriptions on the ratings, respectively. We randomly select 250 video intervals and compare the Rationalization, WAA (λ_a=10, λ_c=100) and SAA (λ_c=100) predictions. The humans are asked to rate a description/explanation on the scale $\{1..4\}$ (1: correct and specific/detailed, 2: correct, 3: minor error, 4: major error). We collect ratings from 3 human judges for each task. Finally, we compute the majority vote, *i.e.*, at least 2 out of 3 judges should rate the description/explanation with a score 1 or 2.

As shown in Table 3, our WAA model outperforms the other two, supporting the results above. Interestingly, Rationalization does better than SAA on this subset, according to humans. This is perhaps because the explanation in SAA relies on the exact same visual evidence as the controller, which may include counterfactually important regions (*i.e.*, there *could be* a stop sign here), but may confuse the explanation module.

Qualitative Analysis of Textual Justifier. As Figure 5 shows, our proposed textual explanation model generates plausible descriptions and explanations, while our model also provides attention visualization of their evidence. In the first example of Figure 5, controller sees neighboring vehicles and lane markings, while explanation model generates "the car is driving forward (description)" and "because traffic is moving freely (explanation)". In Figure 5, we also provide other examples that cover common driving situations, such as driving forward (1st example), slowing/stopping (2nd, 3rd, and 5th), and turning (4th and 6th). We also observe that our explanations have significant diversity, *e.g.*, they provide various reasons for stopping: red lights, stop signs, and traffic. We provide more diverse examples as supplemental materials.

6 Conclusion

We described an end-to-end explainable driving model for self-driving cars by incorporating a grounded introspective explanation model. We showed that (i) incorporation



	1	4
Human:	The car steadily driving + now that the cars are moving.	The car turns left + since there were no oncoming cars.
Ours (WAA):	The car is driving forward + because traffic is moving freely.	The car is turning left + to enter a road.
Ours (SAA):	The car heads down the road + because traffic is moving at a steady pace.	The car turns left + because the road is clear
Rationalization:	The car slows down + because it's getting ready to a stop sign.	The car is slowing down + because it's turning to the left.

	2	5
Human:	The car slows down + since it is about to turn left.	The car is slowing down + in preparation of a turn.
Ours (WAA):	The car slows down + because it is preparing to turn to the road.	The car is slowing down + because the car in front is moving slowly.
Ours (SAA):	The car is slowing + because it is approaching a stop sign.	The car is slowing down + because the car in front has stopped.
Rationalization:	The car slows + because there is a stop sign.	The car slows down + because it's turning to the right.

	3	6
Human:	The car is stopped + while it waits for traffic in front of it to move.	The car is steering to the left and moving forward slowly + the car is negotiating a left hand corner.
Ours (WAA):	The car is stopped + because traffic is stopped.	The car slows down + because it's turning to the left.
Ours (SAA):	The car is stopped + because the car in front of it is stopped.	The car heads down the road + because the traffic is moving at a steady speed.
Rationalization:	The car is stopped + because it is parked in the left lane.	The car is moving forward + because the road is clear

Fig. 5: Example descriptions and explanations generated by our model compared to human annotations. We provide (top row) input raw images and attention maps by (from the 2nd row) vehicle controller, textual explanation generator, and rationalization model (Note: (λ_c, λ_a) = (100,10) and the synthetic separator token is replaced by '+').

of an attention mechanism and prior inputs improves vehicle control prediction accuracy compared to baselines, (ii) our grounded (introspective) model generates accurate human understandable textual descriptions and explanations for driving behaviors, (iii) attention alignment is shown to be effective at combining the vehicle controller and the justification model, and (iv) our BDD-X dataset allows us to train and automatically evaluate our interpretable justification model by comparing with human annotations.

Recent work [11] suggests that causal filtering over attention heat maps can achieve a useful reduction in explanation complexity by removing spurious blobs, which do not significantly affect the output. Causal filtering idea would be worth exploring to obtain causal attention heat maps, which can provide the causal ground of reasoning. Furthermore, it would be beneficial to incorporate stronger perception pipeline, e.g. object detectors, to introduce more "grounded" visual representations and further improve the quality and diversity of the generated explanations. Besides, incorporating driver's eye gaze into our explanation model for mimicking driver's behavior, would be an interesting potential future direction.

Acknowledgements. This work was supported by DARPA XAI program and Berkeley DeepDrive.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. Conference on Learning Representations (2014)
2. Bojarski, M., Choromanska, A., Choromanski, K., Firner, B., Jackel, L., Muller, U., Zieba, K.: Visualbackprop: visualizing cnns for autonomous driving. CoRR, vol. abs/1611.05418 (2016)
3. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L.D., Monfort, M., Muller, U., Zhang, J., et al.: End to end learning for self-driving cars. CoRR abs/1604.07316 (2016)
4. Buehler, M., Iagnemma, K., Singh, S.: The DARPA urban challenge: autonomous vehicles in city traffic, vol. 56. springer (2009)
5. Chen, C., Seff, A., Kornhauser, A., Xiao, J.: Deepdriving: Learning affordance for direct perception in autonomous driving. In: Computer Vision (ICCV), 2015 IEEE International Conference on. pp. 2722–2730. IEEE (2015)
6. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Aistats. vol. 9, pp. 249–256 (2010)
7. Hendricks, L.A., Akata, Z., Rohrbach, M., Donahue, J., Schiele, B., Darrell, T.: Generating visual explanations. In: European Conference on Computer Vision (ECCV) (2016)
8. Hendricks, L.A., Hu, R., Darrell, T., Akata, Z.: Grounding visual explanations. In: European Conference on Computer Vision (ECCV) (2018)
9. Hochreiter, S., Schmidhuber, J.: Lstm can solve hard long time lag problems. In: Advances in neural information processing systems. pp. 473–479 (1997)
10. Hyndman, R., Koehler, A.B., Ord, J.K., Snyder, R.D.: Forecasting with exponential smoothing: the state space approach. Springer Science & Business Media (2008)
11. Kim, J., Canny, J.: Interpretable learning for self-driving cars by visualizing causal attention. Proceedings of the IEEE international conference on computer vision pp. 2942–2950 (2017)
12. Kinga, D., Adam, J.B.: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR) (2015)
13. Lavie, A., Agarwal, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation. pp. 65–72 (2005)
14. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
15. Lee, H., Grosse, R., Ranganath, R., Ng, A.Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: ICML. pp. 609–616. ACM (2009)
16. Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J.Z., Langer, D., Pink, O., Pratt, V., et al.: Towards fully autonomous driving: Systems and algorithms. In: Intelligent Vehicles Symposium (IV). pp. 163–168. IEEE (2011)
17. Lombrozo, T.: Explanation and abductive inference. The Oxford handbook of thinking and reasoning (2012)
18. Lombrozo, T.: The structure and function of explanations. Trends in Cognitive Science **10**(10) (2006)
19. Paden, B., Čáp, M., Yong, S.Z., Yershov, D., Frazzoli, E.: A survey of motion planning and control techniques for self-driving urban vehicles. IEEE Transactions on Intelligent Vehicles **1**(1), 33–55 (2016)
20. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 311–318. Association for Computational Linguistics (2002)
21. Park, D.H., Hendricks, L.A., Akata, Z., Schiele, B., Darrell, T., Rohrbach, M.: Multimodal explanations: Justifying decisions and pointing to the evidence. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

22. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**(1), 1929–1958 (2014)
23. Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Clark, M., Dolan, J., Duggins, D., Galatali, T., Geyer, C., et al.: Autonomous driving in urban environments: Boss and the urban challenge. Journal of Field Robotics **25**(8), 425–466 (2008)
24. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4566–4575 (2015)
25. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: Proceedings of the IEEE international conference on computer vision. pp. 4534–4542 (2015)
26. Xu, H., Gao, Y., Yu, F., Darrell, T.: End-to-end learning of driving models from large-scale video datasets. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2174–2182 (2017)
27. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International Conference on Machine Learning. pp. 2048–2057 (2015)
28. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European Conference on Computer Vision. pp. 818–833. Springer (2014)