SignReasoner: Compositional Reasoning for Complex Traffic Sign Understanding via Functional Structure Units

Supplementary Material

6. SignReasoner Implementations

We elaborate on several implementation details of SignReasoner in this section, including the uniquely designed keys for each FSU (Sect.6.1), the specific prompt and response format employed to instruct the VLM for both Captioning and FSU-Reasoning (Sect.6.2), and the algorithmic implementation of Tree Edited Distance (Sect.6.3) .

6.1. Keys in FSUs

We present the reserved keys for the four Functional Semantic Unit (FSU) categories in Tab. 5.

Table 5. Summary of possible keys in each kind of FSU.

Functions	Possible Keys
Lane	`Turn, Location, Special Lane, Time, Date, Speed, Weight, Height, Other Information`
Direction	`Direction, Via, Destination, Traffic Status, Distance, Other Information`
Construction	`Construction Site, Detour Information, Other Information`
Notice	`Direction, License Plate, Vehicle Type, Time, Date, Road Range, Speed, Weight, Height, Other Information`

6.2. Prompts and Responses

Three kinds of Prompts are adopted in our training data, including Captioning prompt P_{caption}, FSU-Reasoning prompt P_{reason} and the Caption-FSU prompt $P_{\text{cap-FSU}}$, with each corresponding to a unique response format.

Captioning. The captioning prompt (P_{caption}) is designed to instruct VLMs to execute a comprehensive analysis, which entails vividly describing the visual content within traffic sign images, inferring the underlying relations among the sign's various elements, and ultimately articulating the traffic sign's significance and providing actionable driver guidance.

FSU-Reasoning. In contrast, the prompt for FSU-Reasoning (P_{reason}) is primarily structured by providing the model with the predefined FSU categories and corresponding keys, subsequently instructing the model to perform the task of FSU decomposition.

Caption-FSU. Finally, the combined caption-FSU prompt ($P_{\text{cap-FSU}}$) is formulated by integrating the two prior prompt types. The model initially generates a descriptive caption following the instruction of the captioning prompt, and subsequently leverages this intermediate output to guide the FSU-Reasoning task.

The details of these three prompts followed by representative examples of the VLM's generated responses are presented in Tab.6 and Tab.7.

6.3. Tree Edited Distance

This algorithm calculates the Tree Edit Distance (TED), which determines the minimum cost required to transform one tree structure into another using a set of editing operations (insertions, deletions, and modifications). We adopt this algorithm for measuring differences between predicted and ground truth FSU trees. Formally, the detail implementations are illustrated in alg.2. It operates recursively in a bottom-up fashion. For internal nodes, the primary challenge is matching their unordered children. This is solved by constructing a cost matrix based on the recursive distances between all pairs of children and finding the optimal assignment (minimum weight matching) using the Linear Sum Assignment method (i.e., Hungarian Algorithm). For the ordered TED calculation, the assignment process is canceled. The total distance is the sum of the current node's modification cost, the minimum cost of the optimal child assignment, and the costs for handling any unmatched (inserted or deleted) subtrees.

7. Training and Evaluation

This section details the specifics of the training and evaluation datasets (Sect.7.1), the training particulars (Sect.7.2)—including data construction, hyper-parameters, and hardware—and the comprehensive evaluation methodology on the Structure protocols (Sect.7.3).

7.1. Datasets

Training Data. We primarily utilize two types of training data. The first type is the data designated for Caption Supervised Fine-Tuning (SFT). This dataset consists of 5K clear traffic sign images acquired from on-board cameras. We employ the Pangu-MM-38B model for initial annotation, obtaining raw captions which are then subjected to manual

Table 6. Example Prompts and Corresponding Model Responses for SignReasoner.

ID	Prompt (P) / Model Responses
Captioning	$\boldsymbol{P}_{\textbf{caption}}$: Please analyze this cropped traffic sign image in detail and generate a detailed caption describing the colors, shapes, and patterns of the signs, and explain their meanings. This part of the answer should be placed between the <caption> </caption> tags. **Responses:** <caption>This traffic sign is blue in color and has the standard shape of a traffic direction sign, with a white crossroad pattern in the center. The sign displays the names and directions of three roads: 1. Fulong Road (Fulong Rd) is located at the top of the sign, indicating a straight-ahead direction. 2. Mingle Road (Mingle Rd) is on the left side of the sign, indicating a left turn. 3. Yangtaishan Road (Yangtaishan Rd) is on the right side of the sign, indicating a right turn. The color and shape of the traffic sign are designed to be simple and clear, using the white crossroad pattern to clearly indicate the options for going straight ahead or turning left or right. The road names are written in white font, creating a sharp contrast with the blue background, making the information easy to recognize and read. The primary purpose of the traffic sign is to help drivers choose the correct road direction, ensuring safe and efficient travel.</caption>
FSU-Reasoning	$\boldsymbol{P}_{\textbf{Cap-FSU}}$: Please provide the information on the traffic sign in a structured format as follows: 1. Provide the global attribute information of the traffic sign. The output key values include {Traffic Sign, Electronic Sign, Obstruction, Blur, Truncation}, with answers chosen from {Yes, No}. Provide the {Other Global Information} of the traffic sign, with answers selected from the traffic sign itself. 2. Provide the functional category of the traffic sign. The answer should be chosen from {Lane, Direction, Notice, Construction}. 3. Provide the number of functional information items on the sign (for lane information, this refers to the number of lanes). 4. List detailed explanations for each functional information item. For lane information, the output key values include {Electronic Sign, Turn, Location, Special Lane, Time, Date, Speed, Weight, Height, Other Information}. For direction information, the output key values include {Direction, Route, Destination, Traffic Status, Distance, Other Information}. For construction information, the output key values include {Construction Site, Detour Information, Other Information}. For notice information, the output key values include {Direction, License Plate, Vehicle Type, Time, Date, Road Range, Speed, Weight, Height, Other Information}. Only provide answers based on the text and symbols on the traffic sign, without adding any extra explanatory text. Place the structured response within the <FSU> </FSU> tags. **Responses:** <FSU>{"Traffic Sign": "Yes", "Electronic Sign": "No", "Obstruction": "Yes", "Truncation": "No", "Blurriness": "No", "Function Type": "Direction", "Number of Direction Information": "2", "Direction Information 1": {"Direction": "Go Straight", "Via": "Li Yang Road", "Destination": "[The Bund, Haining Road] "}, "Direction Information 2": {"Direction": "Turn Right", "Via": "Li Yang Road", "Destination": "Obstruction"}}</FSU>

quality assurance. The process involved correcting errors in the raw captions and augmenting them with the traffic sign's significance and corresponding driver guidance. The second data type is curated by selecting 726 images from the initial 5K set, ensuring maximum data diversity. This subset was then subjected to a more granular annotation process using the FSU decomposition. Notably, for the first type of data, we mainly adopt it for caption-based SFT, thus all the 5K data are organized in the Captioning prompt formats for training. For the second type of data, due to the synergistic effect as noted in Sect.3.2, we use both the FSU-Reasoning and Caption-FSU formats to generate the training data, resulting in total of 1452 ($i.e.$, 726×2) samples.

TrafficSignEval Benchmark. For evaluation purposes, we establish the first FSU-Reasoning benchmark TrafficSignEval, which comprises 195 distinct traffic sign images, all mutually exclusive from the training data, and is designed to reflect the VLM's understanding performance across each category. Specifically, the benchmark includes 34 Direction signs, 21 Notice signs, 50 Lane signs, and 14 Construction signs. Each traffic sign within the benchmark is also accompanied by the FSU decomposition annotation to facilitate subsequent automated evaluation. All the training and evaluation data are traffic signs cropped from the driving scene images, with the average resolution of 265×225.

Table 7. Example Prompts and Corresponding Model Responses for SignReasoner.

ID	Prompt (P) / Model Responses
Caption-FSU	$P_{\text{FSU-Reason}}$: Please provide the information on the traffic sign in a structured format as follows: 1. Please analyze this cropped image of the traffic sign in detail and generate a detailed caption describing the colors, shapes, and patterns of the signs, and explain their meanings. This part of the answer should be placed between the <caption> </caption> tags. 2. Based on the caption description, provide the global attribute information of the traffic sign. The output key values include {Traffic Sign, Electronic Sign, Obstruction, Blur, Truncation}, with answers chosen from {Yes, No}. Also, provide the {Other Global Information} of the traffic sign, with the answer selected from the traffic sign options. 3. Based on the caption description, determine the functional category of the traffic sign. The answer should be chosen from {Lane, Direction, Notice, Construction}. 4. Based on the caption description, indicate the number of functional information items on the sign (for lane information, this refers to the number of lanes). 5. Based on the caption description, list detailed explanations for each functional information item. For lane information, the output key values include {Electronic Sign, Turn, Location, Special Lane, Time, Date, Speed, Weight, Height, Other Information}. For direction information, the output key values include {Direction, Route, Destination, Traffic Status, Distance, Other Information}. For construction information, the output key values include {Construction Site, Detour Information, Other Information}. For notice information, the output key values include {Direction, License Plate, Vehicle Type, Time, Date, Road Range, Speed, Weight, Height, Other Information}. Only provide the corresponding answers based on the text and symbols on the traffic sign, without adding any extra explanatory text. The structured answers for sections 2-5 should be placed between the <FSU> </FSU> tags. **Responses:** <caption>This traffic sign is blue in color and has a standard traffic sign shape with a white crossroad pattern in the center. The sign displays the names and directions of three roads: 1. Fulong Road (Fulong Rd) is located at the top of the sign, indicating a straight ahead direction. 2. Mingle Road (Mingle Rd) is on the left side of the sign, indicating a left turn. 3. Yangtaishan Road (Yangtaishan Rd) is on the right side of the sign, indicating a right turn. The color and shape of the traffic sign are designed to be simple and clear, using the white crossroad pattern to clearly indicate the options for going straight ahead or turning left or right. The road names are written in white font, creating a sharp contrast with the blue background, making the information easy to recognize and read. The primary purpose of the traffic sign is to help drivers choose the correct road direction, ensuring safe and efficient driving.</caption><FSU>"Traffic Sign": "Yes", "Electronic Sign": "No", "Blocked": "No", "Truncated": "No", "Blurred": "No", "Function Type": "Direction", "Number of Direction Information": "3", "Direction Information 1": "Direction": "Straight Ahead", "Destination": " [Fulong Road, Fulong Rd] ", "Direction Information 2": "Direction": "Left Turn", "Destination": " [Mingle Road, Mingle Rd] ", "Direction Information 3": "Direction": "Right Turn", "Destination": " [Yangtaishan Road, Yangtaishan Rd] "</FSU>

7.2. Training Details

There are mainly two training stages in SignReasoner, the Caption-FSU Distillation and the FSU-GRPO.

In the first stage, we SFT the full parameters of base VLM (i.e., Qwen2.5-VL-7B and Pangu-MM-7B) via the Caption-FSU and FSU-Reasoning formatted SFT data. Both VLMs are trained 20 epochs with the learning rates set to 2×10^{-5}. The min learning rate is set to 2×10^{-6} under cosine decay. The training is implemented on eight 8-GPU nodes with global batch size set to 64. The max encoder sequence length is set to 4096. We follow the ablation results to set the iterative steps as 2 for the best performance.

For the second stage, we adopt the GRPO algorithm for RL training on VLM's full parameters via the Caption-FSU formatted data on eight 8-GPU nodes. The GRPO training uses 1000 iters with learning rate set to 1×10^{-6} and global batch size set to 32. The roll-out number is set to 8, with the roll-out batch size set to 32. For the GRPO hyperparameters, we set gamma to 1.0, lambda to 0.95 and KL coefficient to 0.05. All three rewards are adopted to form the mixed rewards R_{Mixed} as in Eq.6. The TED algorithm is illustrated in Sect. 6.3. We set $\sigma_1 = 0.5, \sigma_2 = 5, \sigma_3 = 0.5$ to stably transform the TED reward into the range [0,1]. The max encoder sequence length is set to 4096.

All the training is implemented under the Ascend 910B3 GPU with PyTorch framework. Details about the GPU have been illustrated in the Compute Reporting Form.

Figure 6. **Visualizations of Traffic Sign Understanding in SignReasoner (Pangu-MM-7B).**

Figure 7. **Visualizations of Traffic Sign Understanding in SignReasoner (Pangu-MM-7B).**

7.3. Evaluation Details

The specific process of automated evaluation in the TrafficSignEval-Structure protocol is illustrated in Fig.8. The input consists of the Predicted and True FSU Dictionaries. First, the matching score, Score_1, for the top-level keys is computed. This score is a weighted sum where p_g is the binary match indicator for the g-th top-level key ($p_g = 1$ for a match, and $p_g = 0$ otherwise). For top-level keys, a match is counted only if their respective values are strictly identical. w_g denotes the weight assigned to each key, with greater weight assigned to critical keys (e.g., `Function` and `FSU Count`, etc.). Score_1 is then compared against a threshold $\epsilon_1(0.8)$: if the score falls below the threshold, the sample is classified as incorrect; otherwise, the evaluation proceeds to the Second-Level Key Score calculation. In this stage, the matching score s_i, for each FSU is firstly calculated. Here, p_{ij} indicates whether the j-th key (out of a total N keys) within the i-th FSU matches. String similarity (with a threshold of 0.5) is employed to determine the match, where $p_{ij} = 1$ for a match and $p_{ij} = 0$ otherwise. Finally, the scores (s_i) of all M FSUs are averaged to yield the Second-Level Key Score_2. If Score_2 falls below the threshold $\epsilon_2(0.5)$, the sample is judged as incorrect; otherwise, it is classified as a correct sample.

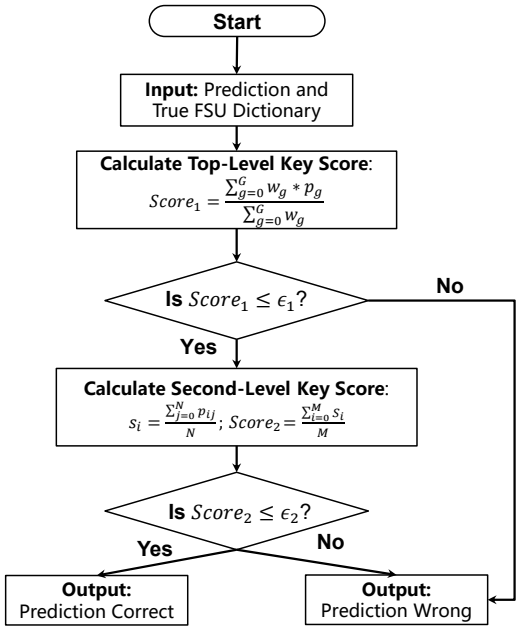

Figure 8. **Flowchart of automatic evaluation algorithm.**

8. More Visualizations

In this section, we provide more comprehensive visualized results (Fig.6-7) in both Chinese and English to demonstrate the powerful capabilities of SignReasoner.

Algorithm 2 Tree Edit Distance

Require: Tree nodes n_1, n_2
Ensure: Minimum edit cost

Function HELPER(n_1, n_2):
$cost \leftarrow 0$
if ISLEAF(n_1) **and** ISLEAF(n_2) **then**
  # Case 1: Both are leaf nodes
  if $n_1.name = n_2.name$ **then**
    if $n_1.value \neq n_2.value$ **then**
      $cost \leftarrow cost + 1$ *# Same name, different value*
    end if
  else
    $cost \leftarrow cost + 2$ *# Different name: replace cost*
  end if
else if ISLEAF(n_1) **then**
  # Case 2: Only n_1 is leaf
  $cost \leftarrow cost + n_2.size + 1$ *# Insert subtree n_2*
else if ISLEAF(n_2) **then**
  # Case 3: Only n_2 is leaf
  $cost \leftarrow cost + n_1.size + 1$ *# Delete subtree n_1*
else
  # Case 4: Both are internal nodes
  if $n_1.name \neq n_2.name$ **then**
    $cost \leftarrow cost + 1$ *# Modify current node name*
  end if
  $C_1 \leftarrow n_1.children, \quad C_2 \leftarrow n_2.children$
  $M \leftarrow |C_1|, \quad N \leftarrow |C_2|$
  Initialize cost matrix D of size $M \times N$
  for $i = 0$ **to** $M - 1$ **do**
    for $j = 0$ **to** $N - 1$ **do**
      $D_{i,j} \leftarrow$ HELPER$(C_1[i], C_2[j])$ *# Recursive calculation*
    end for
  end for
  $(min\_sum, pairs) \leftarrow$ LINEARSUMASSIGNMENT(D) *# Hungarian Algo*
  $cost \leftarrow cost + min\_sum$
  $rows \leftarrow \{r \mid (r, c) \in pairs\}$
  $cols \leftarrow \{c \mid (r, c) \in pairs\}$
  for $i = 0$ **to** $M - 1$ **do**
    if $i \notin rows$ **then**
      $cost \leftarrow cost + C_1[i].size + 1$ *# Unmatched child: delete*
    end if
  end for
  for $j = 0$ **to** $N - 1$ **do**
    if $j \notin cols$ **then**
      $cost \leftarrow cost + C_2[j].size + 1$ *# Unmatched child: insert*
    end if
  end for
end if
return $cost$