Supplementary Material for Hierarchical ConViT with Attention-based Relational Reasoner for Visual Analogical Reasoning

A. Ablation Study on Proposed HCV

The proposed Hierarchical ConViT consists of multiple cascaded ConViT blocks, to capture the visual information from low-level image details to high-level image semantics. To analyze the roles of multiple ConViT blocks, we evaluate each ConViT block (First three rows of results in Table 1) and the combination of ConViT blocks (Last two rows of results in Table 1) on the RAVEN-FAIR dataset. The proposed ARR is used as the reasoning module. From Table 1, we have the following observations:

1. Based on the results of Experiment **1**, **2** and **3**, where the features of the 1st, 2nd and 3rd ConViT blocks only are used for reasoning respectively, a consistent performance improvement can be witnessed when the network goes deeper. From these results, we can observe that in general the deeper ConViT module produces better results in this work.

2. Based on the results of Experiment **1**, **4** and **5**, we can see the improvements on the reasoning accuracy by considering features from deeper ConViT blocks.

3. Based on the results of Experiment **3** and **5**, we can see the additional features of previous receptive fields are advantageous for reasoning, especially for those settings containing complicated pattern combinations such as `2×2Gird`, `3×3Gird` and `Out-InGird`.

B. Visualization

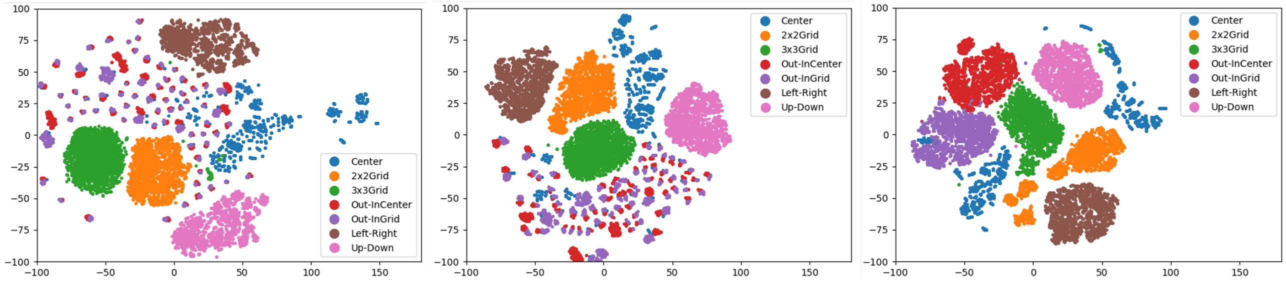
To futher analyze the role of different depth the hierarchical design plays and better understand how the multi-receptive-field design benefits the reasoning results, we perform the t-SNE analysis for different components of the perception module. We take the features from each stage of the proposed HCV and project them using t-SNE to the 2-D space as shown in Fig. 1a, 1b and 1c, respectively.

It can be well noticed from Fig. 1a, 1b that shallow ConViT blocks well separate problem settings such as `2×2Grid`, `3×3Grid`, `Left-Right` and `Up-Down`, but they are unable to discriminate settings that contain "out-in" combinational patterns, *i.e.*, `Out-InCenter` and `Out-InGrid`. When the network goes deeper as shown in Fig. 1c, the perception module can perfectly distinguish all 7 configurations with clear boundaries, but the samples for `Center` are split into several sub-clusters. These observations justify the design choice of utilizing the attentional information of multiple receptive fields.

The potential underlying reason about the performances is that in neural network structures, shallow layers usually capture oriented edges and opponent colors, while deeper layers are specialized to higher-level features and become gradually sparse across feature channels. Therefore, in the proposed HCV-ARR, the hierarchical design is beneficial as shallower ConViT modules are more sensitive to local features of individual objects (lines, edges, colors). When the depth of the network increases, the receptive field increases as well, and effective multi-scale receptive fields can assist

ID	Index of Stages			Accuracy (%) in Different Configurations							
	1st	2nd	3rd	**Avg.**	`Center`	`2×2G`	`3×3G`	`L-R`	`U-D`	`O-IC`	`O-IG`
1.	✓	✗	✗	74.9	83.2	53.1	58.1	90.6	90.3	87.4	61.7
2.	✗	✓	✗	85.8	98.9	66.9	64.5	99.1	99.5	98.5	73.6
3.	✗	✗	✓	93.7	99.8	86.7	84.1	99.6	99.6	99.7	86.9
4.	✓	✓	✗	87.8	98.9	68.1	68.7	99.3	99.2	99.1	78.1
5.	✓	✓	✓	95.4	99.8	92.9	87.9	99.8	99.6	99.7	88.5

Table 1. Ablation study of the proposed HCV on the RAVEN-FAIR dataset.



(a) Features from the 1st stage of HCV.

(b) Features from the 2nd stage of HCV.

(c) Features from the 3rd stage of HCV.

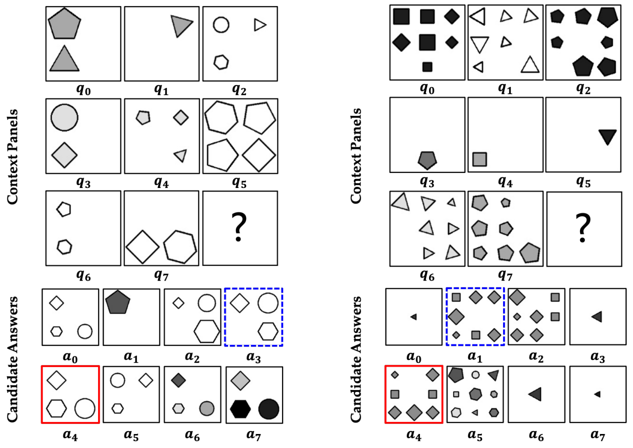
Figure 1. The t-SNE analysis for the proposed HCV.

to recognize patterns with strong spatial and combinational characteristics.

C. Analysis of Failure Cases

It is noticeable that in the configurations containing complicated pattern combinations, *i.e.*, 2×2Gird, 3×3Gird and Out-InGird, the proposed HCV-ARR performs poorer than other settings. We take a step further into these failure cases and analyze the underlying reasons.

For instance, in Fig. 2a, the HCV-ARR fails to induce the correct rule Arithmetic applied on Position, which is to take the binary OR operation of different pattern locations. Meanwhile, it correctly predicts all other relevant attributes, including correct Type, Number and Color.



(a) 2×2Gird

(b) 3×3Gird

Figure 2. Examples of failure cases for the proposed HCV-ARR on the RAVEN-FAIR dataset. The correct answer is framed with red line, while the wrongly predicted answer is framed with blue dotted line.

In Fig. 2b, the rule applied to Position is Progression, which leads to the entities on each panel rolling over the layout. This rule is very different from the Progression rule on other settings, and the HCV-ARR fails to correctly derive such a rule on Position. It is even difficult for humans to induce the reasoning rule and derive the correct answer for these two questions.

D. Model Architecture and Parameters

We provide the detailed architectures and parameters for the proposed Hierarchical ConViT and Attention-based Relational Reasoner as shown in Table 2 and 3, respectively. Please note that in following tables the input is a single RPM image with shape of $(1, 80, 80)$ instead of an entire $(16, 1, H, W)$ RPM panel with 16 images as the notation in the block diagram of main manuscript.

D.1. Hierarchical ConViT

We provide the detailed architectures and parameters for the proposed Hierarchical ConViT as shown in Table 2.

The MHSA in ViT blocks represents the computation of the $\boldsymbol{qkv}$ multi-head self-attention and have no layer operations, which lead to the same dimensionality for the input and the output. Every output feature map marked in **bold** is passed to next successive block, while simultaneously extracted as multi-level receptive fields and transferred into ARR to conduct relational reasoning.

D.2. Attention-based Relational Reasoner

The architectures and parameters for the proposed Attention-based Relational Reasoner are shown in Table 3.

For one input problem panel, the pairwise-attention unit (PairwiseAttn) computes element-wise attentional relations between 3 row/column features in pair and has no layer operations, which gives the corresponding $C_3^2 = 3$ groups of feature tensors with the same dimensionality of input features. A linear layer is applied to aggregate the row/column features, and the dimensionality is reduced by 3.

Stage	Module	Layers	Parameters	Input	Output
1	PatchCrop	Conv2d	C64K4S4	(1,80,80)	
		LayerNorm			(64,20,20)
	ViTBlock×2	MHSA		(64,20,20)	(64,20,20)
		LayerNorm			
		ResBlock	C64		
		Linear	C64		
		LayerNorm			
		ResBlock	C64		**(64,20,20)**
	ConvBlock	Conv2d	C32K7S2P3	(1,80,80)	
		BatchNorm			
		ReLU			(32,40,40)
		Conv2d	C64K3S2P1	(32,40,40)	
		BatchNorm			
		ReLU			**(64,20,20)**
2	PatchMerge	Unfold	C256	(64,20,20)	
		LayerNorm			
		Linear	C128		(128,10,10)
	ViTBlock×2	MHSA		(128,10,10)	(128,10,10)
		LayerNorm			
		ResBlock	C128		
		Linear	C128		
		LayerNorm			
		ResBlock	C128		**(128,10,10)**
	ConvBlock	Conv2d	C64K3S2P1	(64,20,20)	
		BatchNorm			
		ReLU			(64,10,10)
		Conv2d	C128K3S1P1	(64,10,10)	
		BatchNorm			
		ReLU			**(128,10,10)**
3	PatchMerge	Unfold	C512	(128,10,10)	
		LayerNorm			
		Linear	C256		(256,5,5)
	ViTBlock×2	MHSA		(256,5,5)	(256,5,5)
		LayerNorm			
		ResBlock	C256		
		Linear	C256		
		LayerNorm			
		ResBlock	C256		**(256,5,5)**
	ConvBlock	Conv2d	C128K3S2P1	(128,10,10)	
		BatchNorm			
		ReLU			(128,5,5)
		Conv2d	C256K3S1P1	(128,5,5)	
		BatchNorm			
		ReLU			**(256,5,5)**

Table 2. Details of the architectures and parameters for the proposed Hierarchical ConViT, with output channels (C), kernel size (K), stride (S) and padding (P). The sizes of inputs and outputs are given, and the output features at every stage are marked in **bold**.

Stage	Module	Layers	Parameters	Input	Output
1	MatReform	Conv2d	C64K3S1P1	(64,20,20)	
		ResBlock×2	C64		
		Conv2d	C64K3S1P1		
		BatchNorm			(64,20,20)
	AttnReason	PairwiseAttn		(64,20,20)	(3×64,20,20)
		Linear	C64	(3×64,20,20)	
		LayerNorm			(64,20,20)
	DownSample	ResBlock	C128	(64,20,20)	
		BatchNorm			
		AvgPool2d			**(128,1,1)**
2	MatReform	Conv2d	C128K3S1P1	(128,10,10)	
		ResBlock×2	C128		
		Conv2d	C128K3S1P1		
		BatchNorm			(128,10,10)
	AttnReason	PairwiseAttn		(128,10,10)	(3×128,10,10)
		Linear	C128	(3×128,10,10)	
		LayerNorm			(128,10,10)
	DownSample	ResBlock	C128	(128,10,10)	
		BatchNorm			
		AvgPool2d			**(128,1,1)**
3	MatReform	Conv2d	C256K3S1P1	(256,5,5)	
		ResBlock×2	C256		
		Conv2d	C256K3S1P1		
		BatchNorm			(256,5,5)
	AttnReason	PairwiseAttn		(256,5,5)	(3×256,5,5)
		Linear	C256	(3×256,5,5)	
		LayerNorm			(256,5,5)
	DownSample	ResBlock	C128	(256,5,5)	
		BatchNorm			
		AvgPool2d			**(128,1,1)**
Final	MLP	Linear	C256	(3×128,1,1)	
		BatchNorm	C256		
		ReLU			(256,)
	MLP	Linear	C128	(256,)	
		BatchNorm	C128		
		ReLU			(128,)
	MLP	Linear	C1	(128,)	
		Sigmoid			$P(y_i = 1 \mid \boldsymbol{Q}, a_i)$

Table 3. Details of the architectures and parameters for the proposed Attention-based Relational Reasoner with output channels (C), kernel size (K), stride (S) and padding (P). The sizes of inputs and outputs are given, and the output features at every stage are marked in **bold**.