Recurrent Vision Transformer for Solving Visual Reasoning Problems

Nicola Messina(✉), Giuseppe Amato, Fabio Carrara, Claudio Gennaro, and Fabrizio Falchi

Institute of Information Science and Technologies (ISTI), Italian National Research Council (CNR), Via G. Moruzzi 1, 56124 Pisa, Italy
{nicola.messina,giuseppe.amato,fabio.carrara,claudio.gennaro, fabrizio.falchi}@isti.cnr.it

Abstract. Although convolutional neural networks (CNNs) showed remarkable results in many vision tasks, they are still strained by simple yet challenging visual reasoning problems. Inspired by the recent success of the Transformer network in computer vision, in this paper, we introduce the Recurrent Vision Transformer (RViT) model. Thanks to the impact of recurrent connections and spatial attention in reasoning tasks, this network achieves competitive results on the *same-different* visual reasoning problems from the SVRT dataset. The weight-sharing both in spatial and depth dimensions regularizes the model, allowing it to learn using far fewer free parameters, using only 28k training samples. A comprehensive ablation study confirms the importance of a hybrid CNN + Transformer architecture and the role of the feedback connections, which iteratively refine the internal representation until a stable prediction is obtained. In the end, this study can lay the basis for a deeper understanding of the role of attention and recurrent connections for solving visual abstract reasoning tasks. The code for reproducing our results is publicly available here: https://tinyurl.com/recvit.

Keywords: Visual reasoning · Transformer networks · Deep learning

1 Introduction

Deep learning methods largely reshaped classical computer vision, solving many tasks impossible to face without learning representations from data. Convolutional neural networks (CNNs) obtained state-of-the-art results in many computer vision tasks, such as image classification [13,35], or object detection [6,26,27]. Recently, a novel promising architecture took hold in the field of image processing: the Transformer. Initially developed for solving natural language processing tasks, it found its way into the computer vision world, capturing the interest of the whole community. These Transformer-based architectures already proved their effectiveness in many image and video processing tasks [2,7,11,23,24]. The Transformer's success is mainly due to the power of the self-attention mechanism, which can relate every visual token with all the others, creating a powerful relational understanding

S. Sclaroff et al. (Eds.): ICIAP 2022, LNCS 13233, pp. 50–61, 2022.
https://doi.org/10.1007/978-3-031-06433-3_5

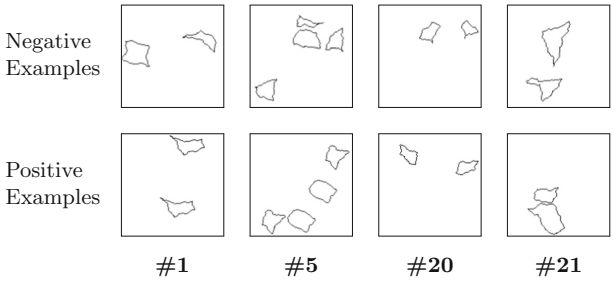


Fig. 1. Positive and negative examples from the considered SVRT problems: P.1: same shapes; P.5: two twisted pairs of same shapes; P.20: same shapes reflected along an unknown symmetry axis; P.21: same shapes but rotated and scaled.

pipeline. In this paper, we aim at studying the relational understanding capabilities of Vision Transformers in the context of an apparently simple yet non-trivial task, called *same-different* task. In short, the same-different task consists in understanding if two shapes in an image satisfy a certain rule. In the simpler case, the rule is merely that *the two shapes must be equal*; however, the rule is not known a priori and must be internally understood from the provided positive and negative examples. An example is given in Fig. 1. Humans perceive the world as a complex set of patterns composite together to form higher-level structures, such as the repeating chorus in a song. Through the same-different task, we can better understand the abstract abilities of current deep neural network models, even outside the computer vision world. The long-term results from these studies can be applied in a wide range of disciplines, from robotics and intelligent video surveillance to cultural heritage preservation.

The same-different task can be framed as a binary classification problem, and it has been partially solved with state-of-the-art convolutional architectures, particularly with ResNets [3,14,22,25]. From these studies, it has been observed that (a) deep CNNs are needed, with lots of free parameters, to relate distant zones of the image in search of matching patterns, and (b) usually, a lot of data is needed to learn the underlying rule, while humans can spot it with only a few samples. Furthermore, some works [18] emphasized the role of recurrent connections, which can iteratively refine the visual input until an optimal and stable conclusion is drawn. In the light of these observations, in this paper, we introduce a novel architecture, called Recurrent Vision Transformer (RViT), for solving the same-different problems. It is inspired by both the recent Vision Transformer (ViT) model [11] and by a recurrent version of the Transformer architecture called Universal Transformer [8]. The introduced architecture can understand and relate distant parts in the image using the powerful Transformer's attentive mechanism and iteratively refine the final prediction using feedback connections. Notably, we find that the base ViT model cannot learn any of the same-different tasks, suggesting that both a hybrid architecture (upstream CNN + downstream Transformer) and feedback connections can be the keys for solving the task.

To summarize, the contribution of the paper is many-fold: (a) we introduce a novel architecture, called Recurrent Vision Transformer (RViT), a hybrid Convolutional-Transformer architecture for solving the challenging same-different tasks; (b) we compare the network complexity and accuracy with respect to other architectures on the same task, obtaining remarkable results with less free parameters and thus better data efficiency; (c) we qualitatively inspect the learned attention maps to understand how the architecture is behaving, and we provide a comprehensive study on the role of the recurrent connections.

2 Related Work

Vision Transformers. The massive engagement of the Transformer architecture [33] in the Natural Language Processing community grew at the point that it trespassed the boundaries of language processing, finding wide applications in computer vision. In fact, it is possible to subdivide images into *patches* which can be fed as input to a Transformer encoder for further processing. Some of the Transformer-based architectures for vision, like Cross Transformers [10] or DETR [4], use the regular grid of features from the last feature map of a CNN as visual tokens. More recently, fully-transformer architectures, first among which ViT [11], have taken root. For the first time, no convolutions are used to process the input image. In particular, the ViT architecture divides the image in patches using the grid approach; the RGB pixel values from every patch are concatenated, and they are linearly projected to a lower-dimensional space to be used as visual tokens. The BERT-like [CLS] token [9] is then used as the classification head. Similarly, the TimeSformer [2] redefined attention both in space and time to understand long-range space-time dependencies in videos.

Same-Different Task. Many tasks have been proposed in computer vision to tackle abstract visual reasoning abilities of machine learning models, like CLEVR and Sort-of-CLEVR [17], Raven's Progressive Matrices (RPM), or Procedurally Generated Matrices (PGMs) [28]. In [12], the authors introduced the *Synthetic Visual Reasoning Test* (SVRT) dataset, composed of simple images containing closed shapes. It was developed to test the relational and comparison abilities of artificial vision systems. The work in [30] first showed, using the SVRT dataset, that the tasks involving comparisons between shapes were difficult to solve for convolutional architectures like LeNet and GoogLeNet [31]. The authors in [20] drawn a similar conclusion, introducing a variation of the SVRT dataset – the Parametric SVRT (PSVRT) for solving some shortcomings of the SVRT dataset – and concluding that the Relation Network [29] is also strained on the same-different judgments. Similarly, [25] developed a more controlled visual dataset to evaluate the reasoning abilities of deep neural networks on shapes having different distributions. The authors in [3,14] found that deep CNNs, like ResNet-50, can solve the SVRT problems even with a relatively small amount of samples (28k images). The authors in [21,22] demonstrated that also many other state-of-the-art deep learning architectures for classifying images (ResNet, DenseNets,

CorNet) models can learn this task, generalizing to some extent. Recently, [32] discussed the important role of attention in the same-different problems.

Recurrent Models. Recurrent models – LSTMs [16] and GRUs [5], to name a few – have been widely used for dealing with variable-length sequences, especially in the field of natural language processing. However, recently, many neuroscience and deep-learning works claimed the importance of recurrent connections outside the straightforward text processing, as they could have an essential role in recognition and abstract reasoning. The work in [18] claimed that the visual cortex could be comprised of recurrent connections, and the visual information is refined in successive steps. Differently, many works in deep learning tried to achieve Turing-completeness by creating recurrent architectures with dynamic halting mechanisms [1,8,15]. Although our work does not include dynamic halting mechanisms, it partially embraces these ideas, experimenting with recurrent connections for iteratively refining the final prediction.

3 The Recurrent Vision Transformer Model

The proposed model is based on the recent Vision Transformer – in particular, the ViT model [11]. The drawback of CNNs in solving the same-different problems is that sufficiently deep networks are needed to correlate distant zones in the image. The Transformer-like attention mechanism in ViT helps in creating short paths between image patches through the self-attention mechanism. Furthermore, inspired by the role of recurrent connections in the human's visual cortex [18], we modify the ViT Transformer encoder module by sharing the encoder weights among all the T layers (i.e., along the depth dimension), effectively creating a recurrent Transformer encoder model, similar to [8]. This has the effect of sharing weights not only in the sequence dimension as in standard Transformers, but also in the depth dimension, further constraining the model complexity. As a feature extractor, we use a small upstream CNN that outputs $N \times N$ $D-$dimensional features used as visual tokens in input to the Transformer encoder. The overall architecture is shown in Fig. 2.

By leveraging the recurrent nature of the architecture, we avoid explicitly tuning the depth of the network (i.e., the total number of recurrent iterations) by forcing the architecture to perform a prediction at each time step, using the CLS token. The most likely outcome among the predictions from all the time steps is then taken as the final prediction. More in detail, the model comprises T binary classification heads, one for each time step. During training, the binary cross-entropy loss at each time step is computed as $\mathcal{L}_t = \mathrm{BCE}(y_t, \hat{y})$, where y_t is the network output from the t-th time step, and $\hat{y}$ is the ground-truth value. The various losses are then aggregated to obtain the final loss $\mathcal{L}_{\text{total}}$. We noticed that a simple average $\frac{1}{T}\sum_{t=1}^{T}\mathcal{L}_{\text{t}}$ already led to good results. However, we obtained the best results by using the automatic loss-weighting scheme proposed in [19]:

$$\mathcal{L}_{\text{total}} = \frac{1}{2}\sum_{t=1}^{T}\left(\frac{1}{e^{s_t}}\mathcal{L}_t + s_t\right), \tag{1}$$

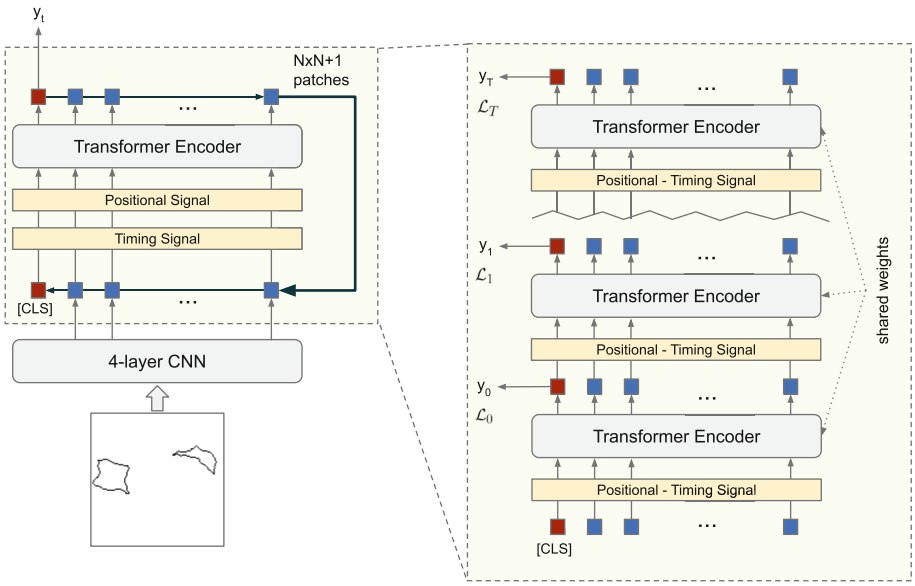


Fig. 2. The RViT architecture. The image is processed by a 4-layer CNN, outputting a 8×8 grid of visual features. The CLS token is added to this set, and the tokens are processed multiple times by the recurrent module. At each time step, the binary cross-entropy loss is computed against the ground-truth labels.

where s_t is a free scalar parameter that encodes the predicted uncertainty of the classification at the t-th time step, and the model automatically learns it during the training phase. We refer readers to [19] for more detailed derivation and discussion.

During inference, the maximum-likelihood prediction is taken as the final network output. In particular, the time step $\bar{t}$ at which the network reaches the maximum confidence is the one where the output probability is farthest from the pure chance in a binary classification setup ($\mathrm{p} = 0.5$):

$$\bar{t} = \arg\max_t |y_t - 0.5|. \tag{2}$$

At this point, the final output is simply $y = y_{\bar{t}}$.

4 Experiments

In this section, we briefly introduce the SVRT dataset used in the experiments, and we present and discuss the performance of the Recurrent Vision Transformer on these problems.

4.1 Dataset

In this work, we use the *Synthetic Visual Reasoning Test* (SVRT) benchmark to test our proposed architecture. SVRT comprises 23 different sub-problems; each sub-problem comprises a set of positive and negative samples generated using a problem-specific rule. The objective of any classifier trained on a problem is to distinguish the positive and negative samples, and the only way to succeed is to discover the underlying rule.

From previous works [3,20] it is clear that relational problems – the ones involving shape comparisons under different geometric transformations – are the most difficult to solve for Deep Neural Networks. Thus, as in [21,22], we focus the attention on four of these problems: **Problem 1 (P.1)** - detecting the very same shapes, randomly placed in the image, having the same orientation and scale; **Problem 5 (P.5)** - detecting two pairs of identical shapes, randomly placed in the image. **Problem 20 (P.20)** - detecting the same shape, translated and flipped along a randomly chosen axis; **Problem 21 (P.21)** - detecting the same shape, randomly translated, orientated, and scaled. Positive and negative samples from each of these visual problems are shown in Fig. 1.

4.2 Setup

For the upstream CNN processing the pixel-level information, we used a 4-layer *Steerable CNN* [34]. A Steerable CNN describes E(2)-equivariant (i.e., rotation- and reflection-equivariant) convolutions on the image plane $\mathbb{R}^2$; in contrast to conventional CNNs, E(2)-equivariant models are guaranteed to generalize over such transformations other than simple translation and are therefore more data-efficient. In the ablation study in Sect. 4.4, we will give more insights on the role of Steerable CNNs over standard CNNs in solving the same-different task.

We forged two different versions of the RViT, a *small* and a *large* version, having the same structure but a different number of hidden neurons in the core layers: the small RViT produces 256-dimensional keys, queries, and values and outputs 256-dimensional visual features from the CNN, while the large RViT has these two parameters set to 512. We used the Adam optimizer; after a minor hyper-parameter tuning, we set the learning rate for all the experiments to 1e–4, and the number of attention heads to 4; we let the models train for 200 epochs, decreasing the learning rate to 1e–5 after 170 epochs. We tested the models using the snapshot with the best accuracy measured on the validation set.

In order to better compare with the ResNet-50 experiments in [3], we also tried to use as up-stream CNN the first two or three layers of a ResNet-50 pre-trained on ImageNet. For the image resolution, we mainly used $N = 16$, outputting 16×16 visual tokens from the CNN. During the pre-training experiments, instead, we used $N = 8$ for accommodating the output feature map resolution of the pre-trained model and also for performance reasons. During training, we set the maximum time steps $T = 9$.

We collected results using both 28k training images, following [3], and 400k training images, for comparing our proposed architectures with convolutional

Table 1. Accuracy (%) of our method, trained from-scratch, with respect to the baselines. #pars indicate the number of free parameters of the model.

Model	400k training samples				28k training samples				#pars ↓
	P.1 ↑	P.5 ↑	P.20 ↑	P.21 ↑	P.1 ↑	P.5 ↑	P.20 ↑	P.21 ↑	
RN [29]	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	0.4M
ViT [11]	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	26M
ResNet-18 [22]	99.2	**99.9**	95.5	96.2	99.2	98.4	93.7	50.0	11M
ResNet-50 [3]	–	–	–	–	95.4	89.9	92.9	72.6	23M
DenseNet-121 [22]	99.6	98.2	94.2	95.1	73.9	54.7	94.4	**85.8**	6.9M
CorNet-S [22]	96.9	96.8	95.0	**96.9**	98.8	97.1	92.3	82.5	52M
RViT-small	**99.9**	99.4	**98.9**	95.7	**99.6**	98.0	93.9	78.6	**0.9M**
RViT-large	**99.9**	99.0	98.8	96.4	**99.6**	**99.3**	**95.3**	77.8	3.1M

Table 2. Accuracy (%) of RViT-small, with the first layers of a ResNet-50 pre-trained on ImageNet, with respect to the full ResNet-50 baseline. In ResNet-50/11 we kept the first 11 layers, while in ResNet-50/23 the first 23.

Model	P.1 ↑	P.5 ↑	P.20 ↑	P.21 ↑	#pars ↓
ResNet-50 [3]	99.5	98.7	98.9	**92.5**	23M
RViT ResNet-50/11	99.6	98.6	94.5	91.6	**2.3M**
RViT ResNet-50/23	**99.7**	**99.7**	**99.4**	85.2	9.5M

networks trained in [21,22]. We used 18k images both for validation and testing. The images were generated with the SVRT original code, available online[1].

4.3 Results

We compared our model with other key architectures: the Relation Network (RN) [29] which by design should be able to correlate distant zones of the image; the Vision Transformer (ViT) [11] which recently achieved remarkable performance on classification tasks, although it is very data-hungry, and some state-of-the-art convolutional models—ResNet18, ResNet50, CorNet-S and DenseNet121—trained on the same task in [3,21,22]. Notably, CorNet-S also implements feedback connections, although it is much more complex, in terms of number of parameters, than our RViT architecture.

Looking at Table 1, we can see how neither the Relation Network nor the ViT converges on the four visual problems, for both 400k and 28k data regimes. The ViT probably needs more architectural inductive biases to understand the rules, while the relational mechanism of Relation Network is probably too simple for understanding the objects in the image and their relationships. Instead, our RViT model can obtain very competitive results on all tasks and on both data regimes,

[1] https://fleuret.org/git-tgz/svrt.

often outperforming the baselines. Noticeably, the RViT-small can learn all the four problems using only 0.9M free parameters, about 8 times fewer parameters than the smallest convolutional network able to solve the task (DenseNet121). This suggests that the model has the correct structure for understanding the visual problems, without having the possibility to memorize the patterns.

In Table 2, we instead report the accuracy of the small RViT model, where the upstream path is pre-trained on the classification task on ImageNet, following the work in [3]. Even in this case, the RViT achieves competitive results, but with much fewer free parameters and using only a slice – the first 11 and 23 layers – of the pre-trained ResNet-50 architecture.

4.4 Ablation Study

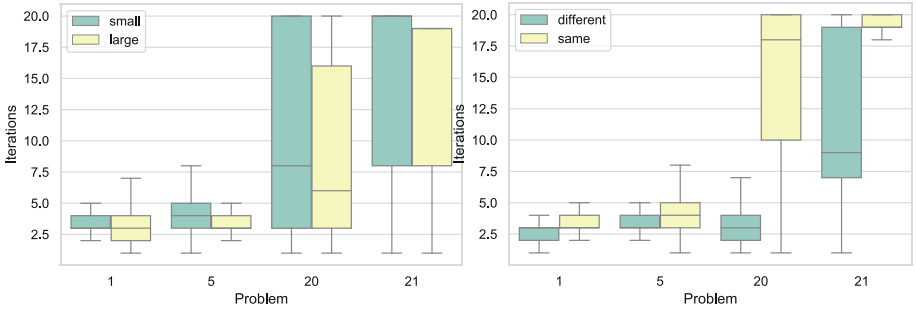
Following, we report some in-depth analysis of the RViTs performed with 28k training images.

The Role of Recurrent Connections and Steerable Convolution. In Table 3, we experimented with some variations of the RViT to understand the roles of recurrent connections and the employed 4-layers steerable CNN. The basic configuration is Conv. ViT, which is the same as the standard ViT from [11] but with an upstream CNN as the visual feature extractor. In contrast to the original ViT formulation, the Conv. ViT can improve significantly on P.1, P.20, and P.21, moving away from the chance accuracy. However, the most significant jump in accuracy happens when recurrent connections are introduced (Conv. RViT). In this case, the same model can learn all the visual problems, with an improvement of 67% on P.1 and 7% on P.21. Another improvement is obtained when using the Steerable CNNs [34]. This kind of CNN produces features equivariant to rotations and reflections. For this reason, it has a wider impact on P.20 and P.21, where shapes are reflected and rotated, respectively.

Recurrent connections seem to have critical importance. They highly regularize the model, making it more data-efficient and performing a dynamic iterative computation that procedurally refines both the previous internal representations and the previous predictions. To better appreciate this aspect, in Fig. 3 we show the mean time step $\bar{t}$, for each problem, where the model reaches the maximum confidence. Interestingly, P.1 and P.5 reach the best confidence in few iterations, while the more challenging P.20 and P.21 need much more pondering before stabilizing. More in detail, it can be noticed that although there is not too much difference considering the size of the models (Fig. 3a), the network seems majorly strained when the shapes are the *same* (Fig. 3b). This is reasonable: it is heavier to be sure that shapes coincide in every point, while it takes little to find even a single non-matching pattern to output the answer *different*.

Table 3. Ablation study on Convolutional ViT (Conv. ViT), on Convolutional Recurrent ViT (Conv. RViT), and Equivariant Convolutional Recurrent ViT (Eq. Conv. RViT). The last one is the model effectively employed in Tables 1 and 2. Accuracy (%) is in this case measured on the validation set.

Model	P.1 ↑	P.5 ↑	P.20 ↑	P.21 ↑
Conv. ViT	59.5	50.0	88.5	62.5
Conv. RViT	99.9	99.0	93.9	66.8
Eq. Conv. RViT	99.8	99.4	95.6	77.3



(a) (b)

Fig. 3. The distribution of the best time step $\bar{t}$ grouped by (a) the two different RViT sizes (small, large), and (b) by the same-different label.

Visualizing the Attention. In Fig. 4, we reported a visualization of the self-attention maps learned by the trained models, computed in specific points (marked with red dots) in the image, and by averaging the four attention heads. The 16×16 grid allows us to appreciate fine details; in particular, we can see what parts of the shapes the model is attending to for producing the final answer. In most cases, the model correctly attends the other shape in search of the corresponding edges. In some instances, the attention map is not so neat (e.g., in (d) and (f)), emphasizing the intrinsic complexity of the tasks. Furthermore, in Fig. 5 we report the evolving attention maps at different time steps. The map is initially very noisy, but it is slowly refined as the number of iterations increases to create a stable representation.

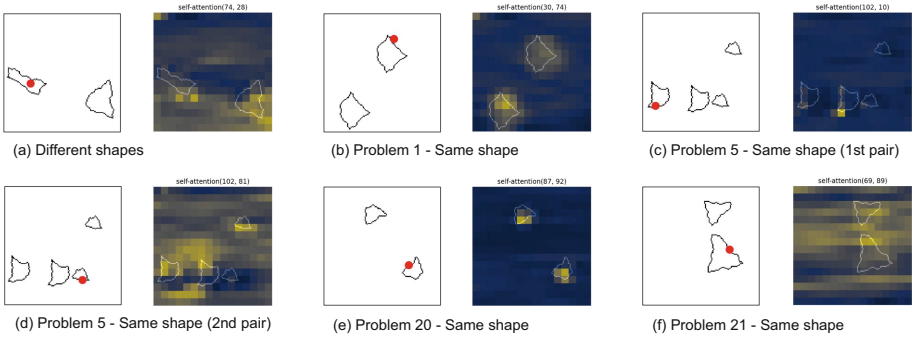


Fig. 4. Attention visualization on the different visual problems. The red dot shows the point in space with respect to which the self-attention is computed. (Color figure online)

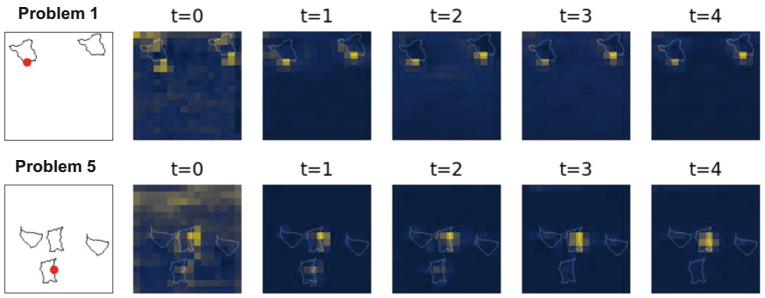


Fig. 5. Evolving attention maps at different time steps.

5 Conclusions

In this work, we leveraged the power of Vision Transformer and recurrent connections to create the Recurrent Vision Transformer Model (RViT) capable of solving some of the most challenging same-different tasks from the SVRT dataset. We showed that this architecture can defeat current methods on the same dataset, while being simpler, more data-efficient, and explainable to some extent. The experiments confirm the hypothesis that recurrent connections provide helps for understanding these visual problems, and the Transformer-like spatial attention enabled us to visualize what parts of the image the model is attending during the inference. The model outperforms the basic ViT model on this task, as well as other relation-aware architectures such as Relation Networks. In the future, we plan to transfer the seeds of this research to real use cases, where multiple possibly distant inputs need to be related and analyzed to draw a conclusion. For example, in surveillance applications, it may be useful to recognize the *same* person across multiple cameras or, in audio processing, recognize repeating patterns in a song for clustering or retrieval.

Acknowledgements. This work was partially supported by "Intelligenza Artificiale per il Monitoraggio Visuale dei Siti Culturali" (AI4CHSites) CNR4C program, CUP B15J19001040004, by the AI4EU project, funded by the EC (H2020 - Contract n. 825619), and AI4Media under GA 951911.

References

1. Banino, A., Balaguer, J., Blundell, C.: PonderNet: learning to ponder. arXiv preprint arXiv:2107.05407 (2021)
2. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? arXiv preprint arXiv:2102.05095 (2021)
3. Borowski, J., Funke, C.M., Stosio, K., Brendel, W., Wallis, T., Bethge, M.: The notorious difficulty of comparing human and machine perception. In: 2019 Conference on Cognitive Computational Neuroscience, pp. 2019–1295 (2019)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 213–229. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_13
5. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
6. Ciampi, L., Messina, N., Falchi, F., Gennaro, C., Amato, G.: Virtual to real adaptation of pedestrian detectors. Sensors **20**(18), 5250 (2020)
7. Coccomini, D., Messina, N., Gennaro, C., Falchi, F.: Combining efficientnet and vision transformers for video deepfake detection. arXiv preprint arXiv:2107.02612 (2021)
8. Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J., Kaiser, Ł.: Universal transformers. arXiv preprint arXiv:1807.03819 (2018)
9. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT 2019, pp. 4171–4186. Association for Computational Linguistics (2019)
10. Doersch, C., Gupta, A., Zisserman, A.: Crosstransformers: spatially-aware few-shot transfer. arXiv preprint arXiv:2007.11498 (2020)
11. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
12. Fleuret, F., Li, T., Dubout, C., Wampler, E.K., Yantis, S., Geman, D.: Comparing machines and humans on a visual categorization test. Proc. Natl. Acad. Sci. **108**(43), 17621–17625 (2011)
13. Foret, P., Kleiner, A., Mobahi, H., Neyshabur, B.: Sharpness-aware minimization for efficiently improving generalization. arXiv preprint arXiv:2010.01412 (2020)
14. Funke, C.M., Borowski, J., Stosio, K., Brendel, W., Wallis, T.S., Bethge, M.: Five points to check when comparing visual perception in humans and machines. J. Vis. **21**(3), 16–16 (2021)
15. Graves, A., Wayne, G., Danihelka, I.: Neural turing machines. arXiv preprint arXiv:1410.5401 (2014)
16. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
17. Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., Girshick, R.: CLEVR: a diagnostic dataset for compositional language and elementary visual reasoning. In: Proceedings of IEEE CVPR, pp. 2901–2910 (2017)

18. Kar, K., Kubilius, J., Schmidt, K., Issa, E.B., DiCarlo, J.J.: Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. Nat. Neurosci. **22**(6), 974–983 (2019)
19. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7482–7491 (2018)
20. Kim, J., Ricci, M., Serre, T.: Not-so-CLEVR: learning same-different relations strains feedforward neural networks. Interface Focus **8**(4), 20180011 (2018)
21. Messina, N., Amato, G., Carrara, F., Falchi, F., Gennaro, C.: Testing deep neural networks on the same-different task. In: 2019 International Conference on Content-Based Multimedia Indexing (CBMI), pp. 1–6. IEEE (2019)
22. Messina, N., Amato, G., Carrara, F., Gennaro, C., Falchi, F.: Solving the same-different task with convolutional neural networks. Pattern Recogn. Lett. **143**, 75–80 (2021)
23. Messina, N., Amato, G., Esuli, A., Falchi, F., Gennaro, C., Marchand-Maillet, S.: Fine-grained visual textual alignment for cross-modal retrieval using transformer encoders. arXiv preprint arXiv:2008.05231 (2020)
24. Messina, N., Falchi, F., Esuli, A., Amato, G.: Transformer reasoning network for image-text matching and retrieval. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 5222–5229. IEEE (2021)
25. Puebla, G., Bowers, J.S.: Can deep convolutional neural networks learn same-different relations? bioRxiv (2021)
26. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
27. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. Adv. Neural. Inf. Process. Syst. **28**, 91–99 (2015)
28. Santoro, A., Hill, F., Barrett, D., Morcos, A., Lillicrap, T.: Measuring abstract reasoning in neural networks. In: International Conference on Machine Learning, pp. 4477–4486 (2018)
29. Santoro, A., et al.: A simple neural network module for relational reasoning. In: Advances in Neural Information Processing Systems, pp. 4967–4976 (2017)
30. Stabinger, S., Rodríguez-Sánchez, A., Piater, J.: 25 years of CNNs: can we compare to human abstraction capabilities? In: Villa, A.E.P., Masulli, P., Pons Rivero, A.J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 380–387. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44781-0_45
31. Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of IEEE CVPR, pp. 1–9 (2015)
32. Vaishnav, M., Cadene, R., Alamia, A., Linsley, D., Vanrullen, R., Serre, T.: Understanding the computational demands underlying visual reasoning. arXiv preprint arXiv:2108.03603 (2021)
33. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
34. Weiler, M., Cesa, G.: General E(2)-equivariant steerable CNNs. In: Conference on Neural Information Processing Systems (NeurIPS) (2019)
35. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500 (2017)