Is a Video worth $n \times n$ Images? A Highly Efficient Approach to Transformer-based Video Question Answering

Chenyang Lyu[†] **Tianbo Ji**[‡*] **Yvette Graham**[¶] **Jennifer Foster**[†]
[†] School of Computing, Dublin City University, Dublin, Ireland
[‡] Nantong University, China
[¶] School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland
chenyang.lyu2@mail.dcu.ie, ygraham@tcd.ie, jennifer.foster@dcu.ie
jitianbo@ntu.edu.cn

Abstract

Conventional Transformer-based Video Question Answering (VideoQA) approaches generally encode frames independently through one or more image encoders followed by interaction between frames and question. However, such schema incur significant memory use and inevitably slow down the training and inference speed. In this work, we present a highly efficient approach for VideoQA based on existing vision-language pre-trained models where we concatenate video frames to a $n \times n$ matrix and then convert it to one *image*. By doing so, we reduce the use of the image encoder from n^2 to 1 while maintaining the temporal structure of the original video. Experimental results on MSRVTT and TrafficQA show that our proposed approach achieves state-of-the-art performance with nearly $4\times$ faster speed and only 30% memory use. We show that by integrating our approach into VideoQA systems we can achieve comparable, even superior, performance with a significant speed up for training and inference. We believe the proposed approach can facilitate VideoQA-related research by reducing the computational requirements for those who have limited access to budgets and resources. Our code is publicly available at https://github.com/lyuchenyang/Efficient-VideoQA for research use.

1 Introduction

Transformer-based Video Question Answering (VideoQA) (Xu et al., 2016; Yu et al., 2018; Xu et al., 2021b; Bain et al., 2021; Lei et al., 2022) approaches relying on large scale vision transformers (Dosovitskiy et al., 2020) have achieved strong performance in recent years. However, such approaches typically encode multiple video frames separately through one or more *Image Encoders* (Lei et al., 2021; Luo et al., 2021; Xu et al., 2021a; Arnab et al., 2021; Zhong et al., 2022) followed by interaction with question representations. This requires significant memory use and inevitably slows down training and inference speed. In order to reduce the computational cost required for modeling video representations from frames, we propose to arrange the frames sampled from one video as a single image. Specifically, we sample n^2 frames from one video and concatenate them as a single image with $n \times n$ grids.

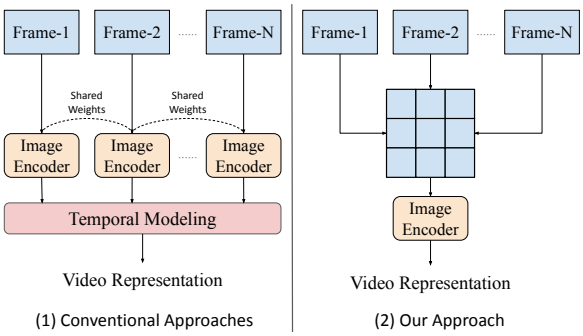

Figure 1: Conventional approach to encoding video frames for VideoQA and our proposed method.

Figure 1 shows the difference between our method (right) and conventional methods (left). In general, conventional approaches either independently encode video frames (Luo et al., 2021; Xu et al., 2021a; Lei et al., 2021; Bain et al., 2022) which require N forward passes, or encode the sequence of all patches in video frames (Bain et al., 2021; Arnab et al., 2021) which quadratically increases the computational cost in the attention modelling. Both types of aforementioned encoding approaches can be expected to negatively impact training and inference speed, whereas our proposed method reduces this need substantially, now requiring only a single forward pass.

Our method diverges from previous approaches in two ways: 1) it fully relies on existing available pre-trained vision-language models such as CLIP (Radford et al., 2021) without need for extra

*corresponding author

Proceedings of The Fourth Workshop on Simple and Efficient Natural Language Processing (SustaiNLP), pages 183–189

pre-training (Bain et al., 2021; Lei et al., 2022); 2) it considers a multi-frame video as a single image, dispensing with the need for positional embedding at the frame level (Bain et al., 2021), so only minor modifications to pre-trained models are necessary.

More importantly, our approach has three advantages: 1) higher computational efficiency[1]; 2) less memory use - our approach only uses an *Image Encoder* a single time; 3) our approach can be easily scaled up for large numbers of frames for long videos. Our approach also models a multiple-frame video as a single image while still (partially) maintaining the temporal structure of the original video.

To validate the effectiveness of our approach, we conduct experiments on two benchmark VideoQA datasets: MSRVTT-MC (Xu et al., 2016; Yu et al., 2018) and TrafficQA (Xu et al., 2021b). Results show that our approach achieves comparable or even superior performance compared to existing models with nearly 4× faster training and inference speed and vast reduction in memory use (30%). Our contribution can be summarised as follows:

- We propose a novel approach combining video frames as a single image to accelerate VideoQA systems ;
- Experimental results on MSRVTT-MC and TrafficQA show that our proposed approach achieves competitive performance with faster training-inference speed and lower memory use;
- We include additional experiments investigating options for arrangement of video frames for VideoQA;

2 Model Architecture

In this section, we introduce details of our approach, of which an overview is shown in Figure 2.

2.1 Vision Transformer

Generally Vision Transformer (ViT) (Dosovitskiy et al., 2020) flattens a single image to m non-overlapping patches $v = \{p_0, p_1, ..., p_{m-1}\}$. All patches are fed into a linear projection and then regarded as discrete tokens in (Vaswani et al., 2017; Radford et al., 2018; Devlin et al., 2019) followed by transformer-based modeling (Vaswani et al., 2017; Dosovitskiy et al., 2020). The output feature is $v^h = \{h_0, h_1, ..., h_{m-1}\}$, where $E^h \in \mathbf{R}^{m \times d}$, d is the dimension of the output feature for each patch.

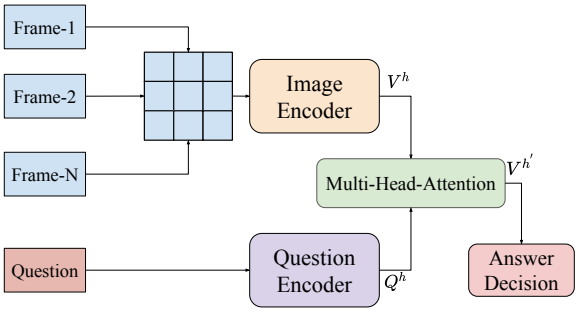

Figure 2: An overview of our proposed approach.

For encoding a multiple-frame video, suppose that we have an input video $V = \{v_0, v_1, ..., v_{n-1}\}$ with n frames, so for each frame v_i ViT flattens it to m non-overlapping patches $v_i = \{p_{i,0}, p_{i,1}, ..., p_{i,m-1}\}$. The patches for each frame are concatenated to form a sequence of patches:

$$V = \{p_{0,0}, ..., p_{0,m-1}, ..., p_{i,0}, p_{i,1}, ..., p_{i,m-1}, ...\} \quad (1)$$

which are fed into (a) linear projection(s) followed by transformer-based modeling (Vaswani et al., 2017; Dosovitskiy et al., 2020)[2]. We thus obtain frame-level representations:

$$V^h = \{v_0, v_1, ..., v_{n-1}\} \quad (2)$$

where $V^h \in \mathbf{R}^{n \times d}$. [3]

2.2 Interaction with Question Representations

For a natural language question $Q = \{w_0, w_1, ..., w_{k-1}\}$ consisting of k words, we use a textual transformer to encode Q to obtain a sentence-level representation $Q^h \in \mathbf{R}^{1 \times d}$. Since in this work, we mainly focus on reducing the computational cost of encoding videos, we perform simple interactions between video representations V^h and question representations Q^h:

[1] For example, the computational cost of Bain et al. (2021); Arnab et al. (2021) scales up quadratically w.r.t. the number of image patches whereas ours is invariant w.r.t. the number of image patches.

[2] Frames can be encoded separately through one or more *Image Encoder* (Luo et al., 2021; Xu et al., 2021a; Bain et al., 2022) or all patches can also be concatenated and passed into one *Image Encoder* (Bain et al., 2021; Arnab et al., 2021)

[3] Patch-level representations $V^h = \{h_{0,0}, ..., h_{0,m-1}, ..., h_{i,0}, h_{i,1}, ..., h_{i,m-1}, ...\}$ are used in (Lei et al., 2021; Bain et al., 2021)

$$V^{h'} = \text{MULTI-HEAD-ATTENTION}(Q^h, V^h, V^h) \quad (3)$$

where $V^{h'} \in \mathbf{R}^{1\times d}$ is the question weighted representations and MULTI-HEAD-ATTENTION (Vaswani et al., 2017) performs attention between V^h (key and value) and Q^h (query).

2.3 Frames Transformation

When encoding multiple frames, the encoding schema in 2.1 incurs significant memory use and additionally impedes training and inference speed (Lei et al., 2021; Luo et al., 2021; Xu et al., 2021a; Lei et al., 2022). Therefore, we propose a novel strategy to reduce the computational cost associated with encoding videos by combining all frames into a single image arranged by $n \times n$ grids. Practically, we arrange all frames to a matrix, M, in which each entry corresponds to a frame. For a video with $n \times n$ frames, we put each frame into $M_{i,j}$ in a specific order. For example, frames can be arranged in M via ascending or descending order (either vertically or horizontally) based on its index in the video.[4] Next, we convert M to a single image. Therefore, regardless of how many frames we use, the number of tokenized patches (image tokens) is always a constant number, resulting in a computationally more efficient VideoQA system.

3 Experiments

3.1 Datasets

We conduct experiments on two benchmark datasets for VideoQA: MSRVTT-MC (Xu et al., 2016; Yu et al., 2018) and TrafficQA (Xu et al., 2021b), which are multi-choice VideoQA datasets – each video in MSRVTT-MC is associated with 5 candidate options whereas TrafficQA provides 4 options for each question. We follow the standard data split for MSRVTT-MC (Xu et al., 2016; Yu et al., 2018), where evaluation data have 2,990 videos. TrafficQA contains 62,535 QA pairs and 10,080 videos. We follow the standard split of TrafficQA: 56,460 QA pairs for training and 6,075 QA pairs for evaluation.

[4]The effect of various arrangement orders is shown in Sec 3.5.

Models	Accuracy
JSFusion (Yu et al., 2018)	83.4
ActBERT (Zhu and Yang, 2020)	85.7
ClipBERT (Lei et al., 2021)	88.2
MERLOT (Zellers et al., 2021)	90.9
VIOLET (Fu et al., 2021)	90.9
VideoCLIP (Xu et al., 2021a)	92.1
All-in-One (Wang et al., 2022)	92.0
Singularity (Lei et al., 2022)	92.1
Ours + MULTI-FRAME	92.1 (1.0×)
Ours + SINGLE-FRAME	92.2 (3.9×)

Table 1: Evaluation results on MSRVTT-MC (Xu et al., 2016; Yu et al., 2018) dataset. Number in bracket indicates the average of training and inference speed (↑), which is evaluated on Nvidia GTX 3090.

3.2 Experimental Setup

We use CLIP ViT-B/16 (Radford et al., 2021) [5] to initialize our IMAGE-ENCODER and TEXT-ENCODER. We evenly sample 9 frames from the videos in MSRVTT-MC and TrafficQA for the main experiment. We train our model for 20 epochs with a learning rate of 1e-6. The training batch size is 16. We use a maximum gradient norm of 1. The optimizer we used is AdamW (Loshchilov and Hutter, 2019), for which the ϵ is set to 1×10^{-8}.

3.3 Evaluation Results

We show the evaluation results on MSRVTT-MC (Xu et al., 2016; Yu et al., 2018) in Table 1. Furthermore, we conduct experiments on TrafficQA (Xu et al., 2021b) and the results are shown in Table 2. We present the results of separately encoding video frames (MULTI-FRAME) as in Figure 1 (left) and our approach that combines multiple video frames into a single image (SINGLE-FRAME). For SINGLE-FRAME, the frames are arranged in a matrix via horizontally descending order. The evaluation results show that our approach SINGLE-FRAME achieves comparable and even improved performance relative to strong baselines including VideoCLIP (Xu et al., 2021a), All-in-One (Wang et al., 2022), Singularity (Lei et al., 2022) and CMCIR (Liu et al., 2022). SINGLE-FRAME obtains a significant speed up (approaching $\times 4$) compared to MULTI-FRAME approach while maintaining competitive performance. The memory use of SINGLE-FRAME is only 30% of MULTI-

[5]https://openai.com/blog/clip/

Models	Accuracy
Q-type (random) (Xu et al., 2021b)	25.0
QE-LSTM (Xu et al., 2021b)	25.2
QA-LSTM (Xu et al., 2021b)	26.7
Avgpooling (Xu et al., 2021b)	30.5
CNN+LSTM (Xu et al., 2021b)	30.8
I3D+LSTM (Xu et al., 2021b)	33.2
VIS+LSTM (Ren et al., 2015)	29.9
BERT-VQA (Yang et al., 2020)	33.7
TVQA (Lei et al., 2018)	35.2
HCRN (Le et al., 2020)	36.5
Eclipse (Xu et al., 2021b)	37.0
ERM (Zhang et al., 2022)	37.1
TMBC (Luo et al., 2022)	37.2
CMCIR (Liu et al., 2022)	38.6
Ours + MULTI-FRAME	39.7 (1.0×)
Ours + SINGLE-FRAME	39.7 (3.8×)

Table 2: Evaluation results on SUTD-TrafficQA (Xu et al., 2021b) dataset. Number in bracket indicates the average of training and inference speed (↑).

FRAME, which are compared on Nvidia GTX 3090. The results on two benchmark datasets have shown the effectiveness of our approach for improving the computational efficiency while maintaining accuracy of VideoQA systems.

3.4 Effect of Number of Frames

We investigate the effect of the number of video frames used by our approach during the training and inference process. The results are shown in Figure 3. We compare the performance of MULTI-FRAME and SINGLE-FRAME for number of frames ranging from 1 to 25[6] in Figure 3. Results show that: 1) Both MULTI-FRAME and SINGLE-FRAME systems can benefit from more video frames; 2) SINGLE-FRAME is capable of achieving comparable and even better performance against MULTI-FRAME; 3) MULTI-FRAME costs much more computational time than SINGLE-FRAME especially when using a large number of video frames. Therefore, our proposed SINGLE-FRAME approach is able to achieve higher efficiency as well as competitive accuracy.

3.5 Effect of Frame Order

We investigate the effect of the arrangement of video frames used to form a single frame. The

[6]For SINGLE-FRAME with a number of frames that is not a square number we up-sample it to the closest square number. For example, a SINGLE-FRAME that deals with 2 frames with index of $\{0, 1\}$, we upsample it to $\{0, 0, 1, 1\}$ as 2×2 images.

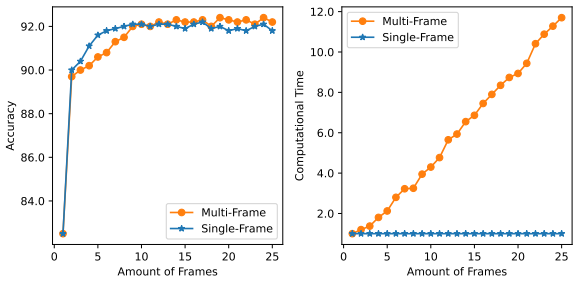

Figure 3: Evaluation results including accuracy (↑) and computational time (↓) on the effect of amount of video frames on MSRVTT-MC.

Arrangement of Frames	9 frames	16 frames
Vertical-Ascent	89.8	89.9
Vertical-Descent	89.1	89.5
Horizontal-Ascent	88.7	88.8
Horizontal-Descent	87.5	88.6
Matrix (Vertical-Ascent)	91.4	91.1
Matrix (Vertical-Descent)	91.9	91.8
Matrix (Horizontal-Ascent)	91.6	91.2
Matrix (Horizontal-Descent)	92.2	92.1
Matrix (Random)	90.5	90.7

Table 3: Evaluation results on the effect of the arrangement of video frames on MSRVTT-MC.

results of both the 9 frame and 16 frame configurations are shown in Table 3. We report the results of VERTICAL, HORIZONTAL and MATRIX via either ASCENDING or DESCENDING order.[7] The results in Table 3 reveal that both VERTICAL and HORIZONTAL perform worst, and this is likely due to configurations distorting the visual information since they essentially squeeze the video frames either vertically or horizontally. The MATRIX arrangement performs substantially better especially MATRIX (HORIZONTAL-DESCENT) and HORIZONTAL generally yielding better performance compared to VERTICAL under the MATRIX arrangement.

4 Conclusion and Future Work

In this paper, we propose a highly efficient method for VideoQA where we combine multiple video frames into one single image. By adapting our approach, the computational cost of VideoQA systems can be significantly reduced. To validate the effectiveness of our approach, we conduct experiments on two benchmark datasets, MSRVTT-MC and TrafficQA. Results show that our approach

[7]Examples are shown in Appendix A.1.

achieves competitive performance and faster training and inference speed (nearly $4\times$ faster) and less memory consumption (30%). Our approach provides a way of significantly accelerating training and inference. In the future, we aim to explore how to adapt our approach to VideoQA with longer videos and additional video-related NLP tasks.

Acknowledgements

This work was funded by Science Foundation Ireland through the SFI Centre for Research Training in Machine Learning (18/CRT/6183). We thank the reviewers for helpful feedback.

Limitations

The two VideoQA datasets used in experiments are associated with relatively short videos. Therefore it would be better if more experiments could be conducted on VideoQA datasets with long videos to verify the effectiveness of our approach on a wider range of VideoQA tasks. Although the proposed approach in this paper can also be used in other video-language tasks, our experiments focuses on a specific video-language task - VideoQA. Experiments on more video-language tasks are needed to show that our approach are also effective in other video-language tasks.

References

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. 2021. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6836–6846.

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738.

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. 2022. A clip-hitchhiker's guide to long video retrieval. *arXiv preprint arXiv:2205.08508*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. 2021. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*.

Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. 2020. Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9972–9981.

Jie Lei, Tamara L Berg, and Mohit Bansal. 2022. Revealing single frame bias for video-and-language learning. *arXiv preprint arXiv:2206.03428*.

Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7331–7341.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. 2018. Tvqa: Localized, compositional video question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1369–1379.

Yang Liu, Guanbin Li, and Liang Lin. 2022. Cross-modal causal relational reasoning for event-level visual question answering. *arXiv preprint arXiv:2207.12647*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*.

Yuanmao Luo, Ruomei Wang, Fuwei Zhang, Fan Zhou, and Shujin Lin. 2022. Temporal-aware mechanism with bidirectional complementarity for video q&a. In *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3273–3278. IEEE.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Mengye Ren, Ryan Kiros, and Richard Zemel. 2015. Exploring models and data for image question answering. *Advances in neural information processing systems*, 28.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Alex Jinpeng Wang, Yixiao Ge, Rui Yan, Yuying Ge, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. 2022. All in one: Exploring unified video-language pre-training.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021a. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6787–6800, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296.

Li Xu, He Huang, and Jun Liu. 2021b. Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9878–9888.

Zekun Yang, Noa Garcia, Chenhui Chu, Mayu Otani, Yuta Nakashima, and Haruo Takemura. 2020. Bert representations for video question answering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1556–1565.

Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 471–487.

Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. 2021. MERLOT: Multimodal neural script knowledge models. In *Advances in Neural Information Processing Systems*.

Fuwei Zhang, Ruomei Wang, Fan Zhou, and Yuanmao Luo. 2022. Erm: Energy-based refined-attention mechanism for video question answering. *IEEE Transactions on Circuits and Systems for Video Technology*.

Yaoyao Zhong, Wei Ji, Junbin Xiao, Yicong Li, Weihong Deng, and Tat-Seng Chua. 2022. Video question answering: Datasets, algorithms and challenges. *arXiv preprint arXiv:2203.01225*.

Linchao Zhu and Yi Yang. 2020. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8746–8755.

A Appendix

A.1 Examples of frames with different arrangement order

We present some examples of frames with different arrangement order in Figure 4, where we use 9 frames as examples. The arrangement orders are: (1) HORIZONTAL. (2) VERTICAL. (3) MATRIX (HORIZONTAL-ASCENT). (4) MATRIX (HORIZONTAL-DESCENT). (5) MATRIX (VERTICAL-ASCENT). (6) MATRIX (VERTICAL-DESCENT). The corresponding video frame indices are shown in Figure 5.

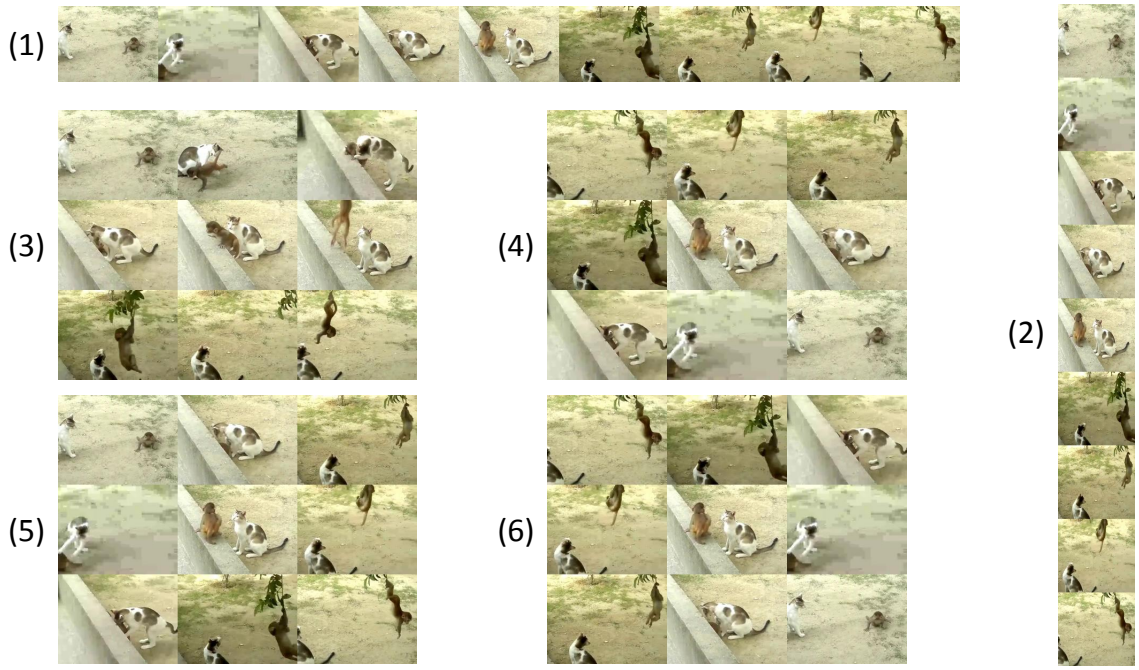

Figure 4: Frames arranged via different order.

(1)

0	1	2	3	4	5	6	7	8

(3)

0	1	2
3	4	5
6	7	8

(4)

8	7	6
5	4	3
2	1	0

(2)

0
1
2
3
4
5
6
7
8

(5)

0	3	6
1	4	7
2	5	8

(6)

8	5	2
7	4	1
6	3	0

Figure 5: Corresponding frame index for different arrangement order.