A Straightforward Framework For Video Retrieval Using CLIP

Jesús Andrés Portillo-Quintero[0000−0002−9856−1900],
José Carlos Ortiz-Bayliss[0000−0003−3408−2166], and
Hugo Terashima-Marín[0000−0002−5320−0773]

School of Engineering and Sciences, Tecnologico de Monterrey
Av. Eugenio Garza Sada 2501, Monterrey, NL 64849, Mexico
a00226024@itesm.mx,
{jcobayliss, terashima}@tec.mx

Abstract. Video Retrieval is a challenging task where the task aims at matching a text query to a video or vice versa. Most of the existing approaches for addressing such a problem rely on annotations made by the users. Although simple, this approach is not always feasible in practice. In this work, we explore the application of the language-image model, CLIP, to obtain video representations without the need for said annotations. This model was explicitly trained to learn a common space where images and text can be compared. Using various techniques described in this document, we extended its application to videos, obtaining state-of-the-art results on the MSR-VTT and MSVD benchmarks.

1 Introduction

Video is one of the most consumed forms of media available on the internet. The high consumption of this type of media requires obtaining suitable methods for finding videos that contain one or more features desired by the users. Most video browsers rely on annotations made by users to identify video contents. Although this solution is simple to implement, it comes at a high price. Relying on annotations to perform a query on videos requires an extensive description of the videos' innards and context. Unfortunately, this information may not be available. Thus, it is clear that a video retrieval system that can handle user's queries without the need for such annotations represents a relevant topic of study.

This document describes a video retrieval model, which, as its name implies, can retrieve the videos from a collection that are best described by a particular query (text). For example, "A woman is running" should return videos that contain women running, thus executing a text-to-video retrieval task (TVR). Given that the video retrieval architecture estimates the similarity between video and text, it can also perform the video-to-text retrieval (VTR) task. It consists of returning captions that best describe the query (video) from a set of description candidates. In either task, the system goal is that, given a query and a set of

video-text pairs, it must return the ranking at which the corresponding opposite modality is positioned.

The TVR and VTR tasks can be seen as methods by which video and text contents are funneled into a fixed-length representation using an embedding function. Since both projections fall in the same dimensional space, a similarity score can be applied, which can be used to rank elements from a set of prospects. Given that similarity metrics between text-video and video-text are equal, TVR and VTR are considered inverse operations. They only depend on the modality of the input prompt.

Some works extensively focus on the video representation by adding pre-trained models considered "experts". Each "expert" focuses on specific video contents such as sound, face detection, motion, among others. The information from all the experts is multiplexed by a complex gating mechanism [5,7]. Instead of starting from an elaborated video representation to train a common visual-text space, we propose to use a learned visual-text space to build a video representation. Similar to Mithun et al. [12], our approach consists of using pre-trained models that measure the similarity between image and text. Then, we extend this idea to handle videos. We experimented with several aggregation methods to comply with the extra temporal dimension.

In this work, we chose Contrastive Language-Image Pretraining (CLIP) as the base image-text model. CLIP is a state-of-the-art neural network, which is pre-trained for image-text pairs [14]. CLIP has proved that similarity learning can be used to train a visual encoder for downstream tasks such as classification, captioning, and clustering, to mention some. We harness the power of its visual representations to create a video representation that can be used directly with its original text encoder to bootstrap a neural network model for video retrieval. Since our work focuses on aggregation strategies of image features, our method is tested with Zero-Shots of the evaluation dataset. Hence, no parameter finetuning is exercised to improve retrieval results.

The remainder of this document is organized as follows. In Section 2 we provide the foundations of this investigation and an overview of the most relevant related works. Section 3 describes the experiments conducted, their main results, and their discussion. Finally, in Section 4, we present the conclusion and some ideas that may be worth exploring as part of the future work.

2 Background and Related Work

The work presented in this document is related to strategies used to construct a video encoder for video retrieval. It is straightforward to think that image features can serve as a proxy for video representations. Karpathy et al. [6] observed that a convolutional neural network (CNN) feature from a single frame could be discriminative enough for video classification, achieving just 1.3 fewer percentage points than the top accuracy model from the same work, which on its part included more visual and temporal information.

Mithun et al. [12] proved that it was possible to supersede the state-of-the-art video retrieval model by obtaining the average visual features obtained from an image-text model. This practice has been implemented on novel models, along with more elaborated video representations. For instance, the state-of-the-art in video retrieval has been pushed by models that implement a Mixture-of-Experts (MoE) paradigm [5,7,10,13]. The MoE approach proposes a complex video representation by multiplexing the outputs of several pre-trained models (known as "experts") that attend to particular aspects of video such as motion, face detection, character recognition, among others.

In this regard, we are aware that at most seven experts have been included in a video retrieval model [5]. Nonetheless, the current state-of-the-art implements a mixture of two experts, indicating that video-text representations may rescind the added complexity that multiple experts convey [13]. Patrick et al. propose that contrastive training used by most video retrieval systems encourages repulsive forces on independent, but similar, examples [13]. To alleviate this, they use a support set containing positive examples for each data point on a training batch, so the common video-text space must learn concept sharing. Nonetheless, contrastive training has been proved successful in image and video representation learning [2,9].

Contrastive training is a regime on which a model is inducted to pull similar data points together and push apart dissimilar ones on a latent space. The foundational mechanism of the Contrastive Language-Image Pretraining (CLIP) is the model used in this work. As the name states, the model is pre-trained on 400,000 image-text pairs collected from the Internet. As a siamese neural network, it is composed of an image (ViT-B/32) and text encoder (transformer) that funnel information into a common space where objects can be compared using cosine similarity [14].

3 Experiment and Results

This section provides a mathematical description of CLIP and how we can use it for VTR or TVR. Later, we describe the datasets and metrics considered for this work. Then, we detail the experiments and their main results, followed by a brief discussion of the most relevant findings.

3.1 CLIP as Video Representation

By using CLIP [14], we obtain the pre-trained functions $\omega(u) = \mathbf{w}$ and $\phi(t) = \mathbf{c_t}$, which encode image u and text t into $\mathbf{w}, \mathbf{c_t} \in \mathbb{R}^d$, where $d = 512$. Assume a video v is composed of s sampled frames such that $v = \{u_1, u_2, \ldots, u_s\}$. Consequently, we can calculate the embedding of each frame into a matrix $\mathbf{W} \in \mathbb{R}^{d \times s}$ so $\mathbf{W} = [\omega(u_1) = \mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_s}]$. Therefore, the problem we try to solve is to find an aggregation function Λ that maps the input $\mathbf{W} \in \mathbb{R}^{d \times s}$ into a video representation $\mathbf{c_v} \in \mathbb{R}^d$. Then, with a video and text representations $\mathbf{c_v}$ and

$\mathbf{c_t}$, we can compute a cosine similarity function (Equation 1), which is useful for ranking the video-text pairs inside a dataset given a query of a specific modality.

$$sim(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|} \tag{1}$$

3.2 Datasets

The proposed framework assumes a set $\mathcal{C}$ of videos and corresponding captions pairs in the form $\{\{(v_i, t_{ij})\}_{i=1}^{n}\}_{j=1}^{m(v_i)}$ where the number of captions per video may be non-uniform, hence m is a function of v. By design, some datasets are split into sections used for training and validation of results. We use the training splits to prove our hypothesis for the preliminary experiments, but final results are reported on tests split of their respective datasets.

The datasets involved in this work are listed below.

MSR-VTT is a dataset composed of 10,000 videos, each with a length that ranges from ten to 32 seconds and 200,000 captions. The training, validation and test splits are composed of 6,513, 497 and 2,990 videos, respectively, with 20 corresponding descriptions each [18]. The test set has been used in different ways in the literature. Then, we will refer to two common variations as Full [7] (containing all the 2,990 videos in the test set from MSR-VTT) and 1k-A [19] (containing only 1,000 videos from the 2,990 in the test set in MSR-VTT).

MSVD contains 1,970 videos, each with a length that ranges from one to 62 seconds. Train, validation and test splits contain 1,200, 100 and 670 videos, respectively [1]. Each video has approximately 40 associated sentences in English.

LSMDC is comprised 118,081 videos, each with a length that ranges from two to 30 seconds. The videos were extracted from 202 movies. The validation set contains 7,408 videos, and the test set 1,000 videos from movies independent from the training and validation splits [15].

All the frames were sampled from each video from the previously mentioned datasets to extract the frame features. Other datasets are related to this work but cannot be used include WIT (WebImageText) [14] and HT100M [11]. WIT is composed of 400,000 image-text pairs on which CLIP was trained on. Since WIT is an image-text dataset that cannot be used as a benchmark for video retrieval. HT100M is a dataset of 100 million video-text pairs, used only as a pre-training set for other Video Retrieval works [5,11,13,16].

3.3 Metrics

To conduct our experiments, we follow the testing methodologies used in previous works [5,7] and report standard retrieval metrics. For median rank (MdR), mean rank (MnR), and standard deviation of rank (StdR), the lower the value, the

better the performance. In the case of recall at rank ($R@k$, where $k = \{1, 5, 10\}$), the higher the value, the better the performance. For datasets that involve multiple sentences per video —such as Full from MSR-VTT and MSVD test—, we follow the protocol used by Liu et al. [7] and use the minimum rank among all associated sentences to a given video query.

3.4 Exploratory Experiments

In the exploratory experiments, we empirically define two candidates for frame-level aggregation Λ functions. We conduct this set of preliminary experiments on a validation sample comprised of 1,000 video-text pairs from MSR-VTT. The first frame-level aggregation function is based on the idea that it is feasible to obtain reasonable video representations by only considering one frame sample [6]. Given the feature matrix $\mathbf{W} \in \mathbb{R}^{d \times s}$, we define $\Lambda_s(W) = W_{30} \in \mathbb{R}^d$ as a function that returns the features of the 30^{th} frame. Since these videos contain approximately 30 frames per second, this is equivalent to sampling a frame from the first second of the video.

A second candidate for an aggregation function is proposed by Mithun et al. [12], who suggest that the average of frame-level features from videos can be used as an approximator for video representations. This method has extensively been used in other retrieval-related works [5,7,9,11,13]. Consequently, we define $\Lambda_{avg}(\mathbf{W}) = \bar{W} \in \mathbb{R}^d$, where $\bar{W}$ is the average value of matrix columns.

Given that videos present dynamic events in which several sequences of frames can represent completely different things, we used k-means as the method for aggregation [17]. With this implementation, the aggregation function follows the form $\Lambda_k(\mathbf{W}) = W \in \mathbb{R}^{d \times k}$, which returns k video embeddings. For evaluation purposes, we repeat the ranking procedure with the obtained independent video representations k times and register each query's minimum rank, then calculate the retrieval metrics.

Λ	**R@1**	**R@5**	**R@10**	**MdR**	**MnR**	**StdR**
Λ_s	24.9	46.1	56.9	7.0	64.61	149.21
Λ_{avg}	35.4	58.0	67.2	3.0	39.81	111.43
Λ_2	34.3	57.8	66.5	3.0	40.23	112.85
Λ_3	34.4	57.7	66.6	3.0	39.77	110.69
Λ_4	33.7	58.4	66.9	3.0	37.98	107.53
Λ_5	34.4	57.6	66.1	3.0	38.44	108.02
Λ_6	34.9	58.4	67.6	3.5	37.44	108.34
Λ_7	35.3	58.1	67.5	4.0	38.33	107.88
Λ_8	33.9	57.7	67.9	3.0	38.23	107.32
Λ_9	33.9	57.2	67.1	3.0	37.87	108.23
Λ_{10}	35.0	57.8	68.0	3.0	37.26	107.34

Table 1: Text-to-video tetrieval results on the MSR-VTT validation set, using various aggregation functions.

Based on the results depicted in Table 1, the average-based methods obtain the best results in terms of the metrics used. It is noticeable that, among k-means methods, there is no significant difference between the results. This may be because MSR videos do not exceed 32 seconds in length, which may not be enough to differentiate the centroids when creating the clusters. We appeal to Occam's Razor principle regarding the aggregation method and select Λ_{avg} for further experiments since it accomplishes a similar performance to k-means based aggregation methods, but with a lower calculation complexity.

3.5 Confirmatory Experiments

This section compares our video retrieval model against the state-of-the-art results for MSR-VTT, MSVD, and LSMDC datasets. In all the cases, we evaluate both the TVR and VTR tasks.

In MSR-VTT, we can supersede the R@1 score of the previous best model SSB [13] on the split 1k-A for the TVR task. However, we are positioned behind previous works on other recall metrics (Table 2). Besides, we consistently achieve state-of-the-art results on all the recall metrics in the Full split from MSR-VTT. In the MSVD dataset, we obtain state-of-the-art results on most of the retrieval metrics (Table 3). We suppose that models based on MoE, such as SSB [13] and CE [7], cannot use all of their implemented experts because the videos in MSVD lack audio information, so they have to rely exclusively on visual features. In LSMDC, we do not obtain state-of-the-art results, but we are positioned second-best (Table 4). Given that video descriptions in this dataset do not follow the form of a typical sentence, as they are designed to teach a model to recognize characters and interactions between movie scenes, we commend the robustness of CLIP's text encoder because it could adapt to a new sentence schema.

3.6 Discussion

Although we obtain outstanding results on different metrics and datasets, there are some things worth discussing. For example, our original supposition was that the ranking worsens as the video gets longer. To confirm or reject this idea, we conducted an experiment on set 1k-A from MSR-VTT (Figure 1). Figure 1a depicts the video length in seconds (x-axis), the rank assigned to it (y-axis), the overall median is a red line, and the average rank is depicted as a blue line. As a video gets longer, we expected that it would be more difficult for the video representation to capture the temporal elements. Hence it would be ranked worse. However, it shows that ranking varies wildly from video length. Notice, there is a possible trend downwards. We claim that pattern results from a bigger sample size on shorter videos that allow for more outliers to appear (Figure 1b). Also, in Figure 1c we observe that, by grouping videos by length, there is no noticeable trend on rank (at least for videos present in the dataset).

We proceeded to look at the worst-ranked video-text pairs. We noticed that several sentences incorporated phrases like "a family is having a conversation" or "a man talking about a woman", hinting that sentences that were mainly

describing audio content would be ranked worse. This conclusion is reinforced by the fact that our model scored the best on MSVD, a dataset that by design does not contain any audio track, and text descriptions are based on what can be visualized.

			TVR				VTR			
Method	Training	Test Set	**R@1**	**R@5**	**R@10**	**MdR**	**R@1**	**R@5**	**R@10**	**MdR**
JSFusion [19]	M	1k-A	10.2	31.2	43.2	13	-	-	-	-
HT100M [11]	H+M	1k-A	14.9	40.2	52.8	9	16.8	41.7	55.1	8
CE [7]	M	1k-A	20.9	48.8	62.4	6	20.6	50.3	64	5.3
AVLnet [16]	H+M	1k-A	27.1	55.6	66.6	4	28.5	54.6	65.2	4
MMT [5]	H+M	1k-A	26.6	57.1	**69.6**	4	27.0	57.5	69.7	3.7
SSB [13]	H+M	1k-A	30.1	**58.5**	69.3	**3**	**28.5**	**58.6**	**71.6**	**3**
CLIP	W	1k-A	**31.2**	53.7	64.2	4	27.2	51.7	62.6	5
VSE [12]	M	Full	5.0	16.4	24.6	47	7.7	20.3	31.2	28
VSE++ [12]	M	Full	5.7	17.1	24.8	65	10.2	25.4	35.1	25
Multi Cues [12]	M	Full	7.0	20.9	29.7	38	12.50	32.10	42.4	16
W2VV [3]	M	Full	6.1	18.7	27.5	45	11.8	28.9	39.1	21
Dual Enc. [4]	M	Full	7.7	22.0	31.8	32	13.0	30.8	43.3	15
E2E [9]	M	Full	9.9	24.0	32.4	29.5	**-**	**-**	**-**	**-**
CE [7]	M	Full	10.0	29.0	42.2	16	15.6	40.9	55.2	8.3
CLIP	W	Full	**21.4**	**41.1**	**50.4**	**10**	**40.3**	**69.7**	**79.2**	**2**

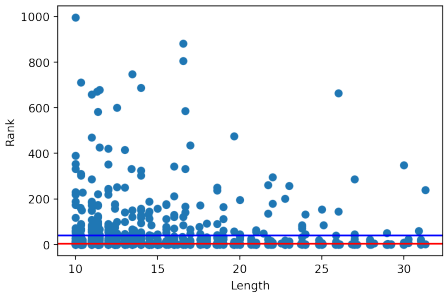
Table 2: TVR and VTR results in the MSR-VTT dataset. M, H and W denote training on MSR-VTT, HT100M and WIT, respectively.

		TVR				VTR			
Method	**Training**	**R@1**	**R@5**	**R@10**	**MdR**	**R@1**	**R@5**	**R@10**	**MdR**
VSE [12]	D	12.3	30.1	42.3	14	34.7	59.9	70.0	3
VSE++ [12]	D	15.4	39.6	53.0	9	-	-	-	-
Multi Cues [12]	D	20.3	47.8	61.1	6	-	-	-	-
CE [7]	D	19.8	49.0	63.8	6	-	-	-	-
Support-set Bottleneck [13]	H+D	28.4	60.0	72.9	4	-	-	-	-
CLIP	W	**37**	**64.1**	**73.8**	**3**	**59.9**	**85.2**	**90.7**	**1**

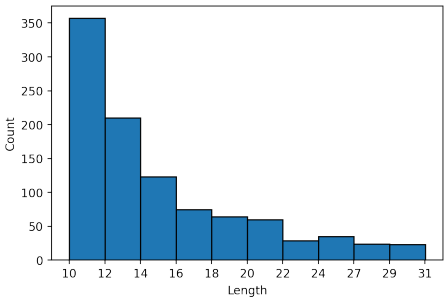
Table 3: TVR and VTR results in the MSVD dataset. D, H and W denote training on MSVD, HT100M and WIT, respectively.

		TVR				VTR			
Method	**Training**	**R@1**	**R@5**	**R@10**	**MdR**	**R@1**	**R@5**	**R@10**	**MdR**
JSFusion [19]	L	9.1	21.2	34.1	36	**12.3**	**28.6**	**38.9**	**20**
CE [7]	L	11.2	26.9	34.8	25.3	-	-	-	-
MMT [5]	H+L	**12.9**	**29.9**	**40.1**	**19.3**	-	-	-	-
CLIP	W	11.3	22.7	29.2	56.5	6.8	16.4	22.1	73

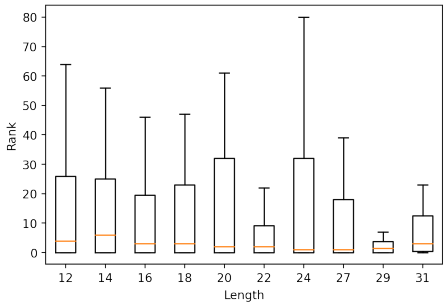
Table 4: TVR and VTR results in the LSMDC dataset. L, H and W denote training on LSMDC, HT100M and WIT, respectively.



(a) Scatter plot of video length and assigned rank.



(b) Histogram of video length.



(c) Distribution of assigned rank per video length groups.

Fig. 1: Analysis on relation of video length and assigned rank on TVR task using the 1k-A test splits.

4 Conclusion and Future Work

This work presents the first implementation of CLIP to obtain video features. Our method works by leveraging its learned common image-text space without the need for parameter finetuning (Zero-Shot). We apply an aggregation function to frame-level features, common in other video retrieval works. Our work focuses only on visual and text modalities, as it supersedes methods that implement a

complex mixture of pre-trained models obtaining state-of-the-art results on the MSVD and MSR-VTT datasets [1].

One potential application of this CLIP-derived implementation is to retrieve specific moments inside videos. Also, it is yet unseen how will our video representation behave if tested as a video classifier. This methodology might be helpful to create a video representation that is based on CLIP for longer durations. For example, other works have used frame features to construct a graph that can change through time [8]. Such a representation could keep the strong text alignment suitable for video retrieval. Also, our work can be used as an expert on a future MoE video retrieval system.

Acknowledgments

This research was partially supported by ITESM Research Group with Strategic Focus on Intelligent Systems.

References

1. Chen, D., Dolan, W.B.: Collecting highly parallel data for paraphrase evaluation. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. pp. 190–200 (2011)
2. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
3. Dong, J., Li, X., Snoek, C.G.M.: Predicting visual features from text for image and video caption retrieval. IEEE Transactions on Multimedia **20**(12), 3377–3388 (2018)
4. Dong, J., Li, X., Xu, C., Ji, S., He, Y., Yang, G., Wang, X.: Dual encoding for zero-example video retrieval. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9346–9355 (2019)
5. Gabeur, V., Sun, C., Alahari, K., Schmid, C.: Multi-modal transformer for video retrieval. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 214–229. Springer International Publishing, Cham (2020)
6. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1725–1732 (2014)
7. Liu, Y., Albanie, S., Nagrani, A., Zisserman, A.: Use what you have: Video retrieval using representations from collaborative experts. In: BMVC (2019)
8. Mao, F., Wu, X., Xue, H., Zhang, R.: Hierarchical video frame sequence representation with deep convolutional graph network. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops. pp. 0–0 (2018)
9. Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-end learning of visual representations from uncurated instructional videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9879–9889 (2020)

[1] The code is publicly available at: `https://github.com/Deferf/CLIP_Video_Representation`

10. Miech, A., Laptev, I., Sivic, J.: Learning a text-video embedding from incomplete and heterogeneous data (2020)
11. Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2630–2640 (2019)
12. Mithun, N.C., Li, J., Metze, F., Roy-Chowdhury, A.K.: Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In: Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval. pp. 19–27 (2018)
13. Patrick, M., Huang, P.Y., Asano, Y., Metze, F., Hauptmann, A.G., Henriques, J.F., Vedaldi, A.: Support-set bottlenecks for video-text representation learning. In: International Conference on Learning Representations (2021)
14. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021)
15. Rohrbach, A., Rohrbach, M., Tandon, N., Schiele, B.: A dataset for movie description. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3202–3212 (2015)
16. Rouditchenko, A., Boggust, A., Harwath, D., Joshi, D., Thomas, S., Audhkhasi, K., Feris, R., Kingsbury, B., Picheny, M., Torralba, A., Glass, J.: Avlnet: Learning audio-visual language representations from instructional videos (2020)
17. Sun, C., Myers, A., Vondrick, C., Murphy, K., Schmid, C.: Videobert: A joint model for video and language representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7464–7473 (2019)
18. Xu, J., Mei, T., Yao, T., Rui, Y.: Msr-vtt: A large video description dataset for bridging video and language. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5288–5296 (2016)
19. Yu, Y., Kim, J., Kim, G.: A joint sequence fusion model for video question answering and retrieval. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 471–487 (2018)