Sports Video Analysis on Large-Scale Data

1、Motivation

过往的video captioning的工作主要面临着三个限制：1）繁重的人工标注工作限制了数据集的大小；2）过往的工作没有开源数据集；3）现有方法在推理的时候经常忽略人们看视频时真正感兴趣的部分。

基于这三个limitation，作者收集了一个更大、更全的数据集NSVA。可以用作video captioning、finegrained action recognition和video re-id

2、NSVA

Datasets	Domain	#Videos	#Sentences	#Hours	Avg. words	Accessibility	Scalability	Multi-task
SVN [54]	basketball	5,903	9,623	7.7	8.8	✗	✗	✗
SVCDV [40]	volleyball	4,803	44,436	36.7	-	✗	✗	✗
NSVA	basketball	**32,019**	**44,649**	**84.8**	6.5	✓	✓	✓

Table 1: The statistics of NSVA and comparison to other fine-grained sports video captioning datasets.

Videos				Sentences				Games				Teams	Actions	Identities
train	val	test	total	train	val	test	total	train	val	test	total	all-sets	all-sets	all-sets
24k	3.9k	3.9k	32k	33.6k	5.5k	5.5k	44.6k	100	16	16	132	10	172	184

Table 2: Data split detail of our dataset.

3、Architecture

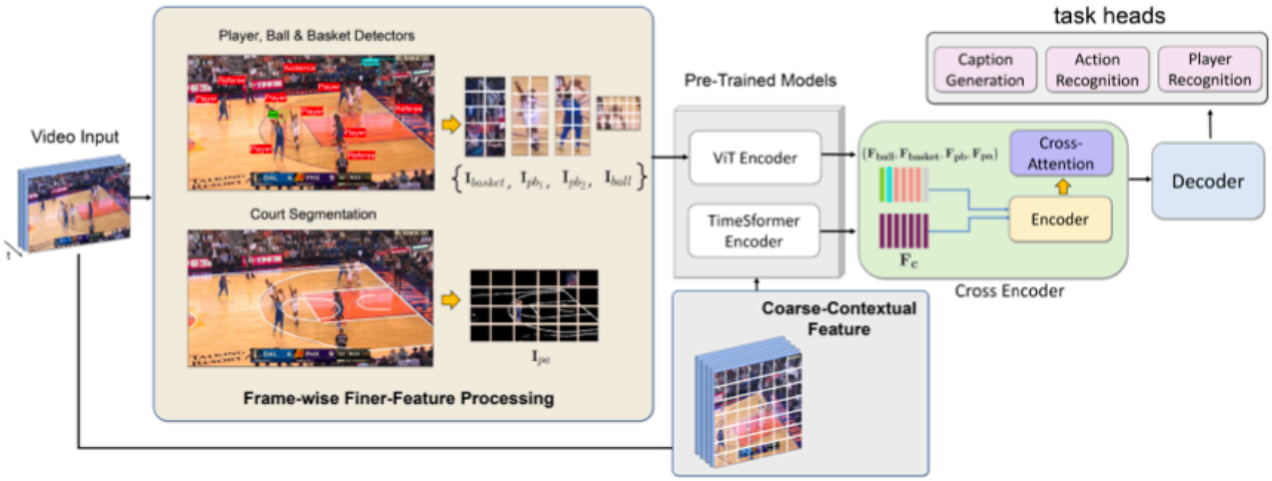

Fig. 2: Pipeline of our proposed approach for versatile sports video understanding. First, raw video clips (left) are processed into two types of finer visual information, namely object detection (including ball, players and basket), and court-line segmentation, all of which are cropped, grided and channelled into a pre-trained vision transformer model for feature extraction. Second, these heterogeneous features are aggregated and cross-encoded with the global contextual video representation extracted from TimeSformer (middle). Third, a transformer decoder is used with task-specific heads to recursively yield results, be it as video captions, action recognition or player identification (right).

这里作者用TimeSformer提取视频特征，用ViT提取球、篮框和运动员和三分线的特征，得到这些特征以后通过一个Transformer encoder进一步提取信息：

$$\mathbf{F}_f = \text{CONCAT}(\text{SUM}(\mathbf{F}_{ball}, \mathbf{F}_{basket}, \mathbf{F}_{pb}), \mathbf{F}_{pa})$$

The overall encoding process is given as

$$\mathbf{V}_c = \text{Transformer}\left(\mathbf{F}_c\right), \mathbf{V}_f = \text{Transformer}\left(\mathbf{F}_f\right),$$

在此之后，使用一个Transformer构建整个场景和细粒度patch之间的关系(cross attention)：

$$M = Transformer(CONCAT(V_c, V_f))$$

最后，用一个Transformer Decoder生成caption。当然也可以将decoder用于其他任务。