Video Co-summarization: Video Summarization by Visual Co-occurrence

Wen-Sheng Chu[1], Yale Song[2] and Alejandro Jaimes[2]

[1]Carnegie Mellon [2]YAHOO! LABS

CVPR 2015 BOSTON JUNE 7-12

Problem

- Video summarization

△Summarize a video of Surfing (eg, a query string):

- Online videos are humongous
  - △82.5% US Internet users view online videos (June 2013)
  - △6B hours of videos viewed per month (Oct 2014)
  - △44B videos view online per month (June 2013)
  - △300+ hours uploaded per minute (May 2015)

source

- Supervised video summarization

Domain knowledge

Sports


Fleischman et al. [ACMMM07]


Chen & Vleechouwer [TCSVT11]


Zhu et al. [ACMMM07]


Kawamura [SIGGRAPH14]

News

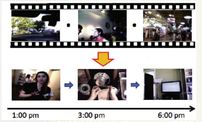

Wu et al. [SPM06], Liu et al. [ACMMM12], Daneshi et a. [ICME13]

Surveillance

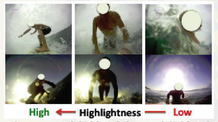

Synopsis: Pritech et al. [TPAMI08]


Video condensation: Feng et al. [CVPR12]

Learning-based Ego. cosumer



Lee et al. [CVPR12], Lu & Grauman [CVPR13]



Potapov et al. [ECCV14], Sun et al. [ECCV14]

- Unsupervised video summarization

△Require no prior knowledge and annotated data
  - Dictionary learning: Cong et al. [TMM12], Zhao & Xing [CVPR14]
  - Hierarchical clustering: Mahmoud et al. [ICMLA13]

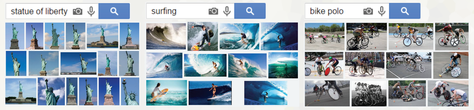
△Use additional resources
  - Human attention during viewing videos: Ngo et al [TCSVT05]
  - Web image priors: Khosla et al. [CVPR13], Kim et al. [CVPR14]

Video Co-summarization

- A desirable video summarization method
  - ☑Generate adaptive summaries that fits user's interests
  - ☑Scale to large dataset
  - ☑Require limited/no human supervision

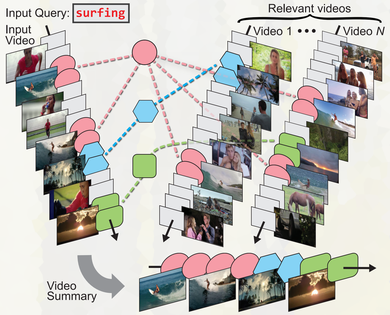
- Observations

△Videos can share common topics (eg, retrieved by a query string)
△Important concepts are likely to repeat visually.



- Main idea

△Exploits visual co-occurrence across multiple videos



- Shot segmentation

- Visual co-occurrence as bipartite graph and co-clusters [1]

1. Compute $\mathbf{D}_1 = \mathrm{diag}(\mathbf{C}\mathbf{e}_n)$
2. Compute $\mathbf{D}_2 = \mathrm{diag}(\mathbf{C}^\top \mathbf{e}_m)$
3. Compute $\widehat{\mathbf{C}} = \mathbf{D}_1^{-1/2}\mathbf{C}\mathbf{D}_2^{-1/2}$
4. Perform clustering on $\mathbf{Z} = [\mathbf{D}_1^{-1/2}\mathbf{U}; \mathbf{D}_2^{-1/2}\mathbf{V}]$ where $\widehat{\mathbf{C}} = \mathbf{U}\Sigma\mathbf{V}^\top$

[1] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In SIGKDD, 2001.

Maximal Biclique Finding (MBF)

- Visual co-occurrence as maximal bicliques

△Co-clusters could fail when shot co-occurrence is sparse.
△We formulate co-sum as a maximal bicluque finding problem:

$$\max_{\mathbf{u},\mathbf{v}} \ \sum_{ij} C_{ij} u_i v_j$$
$$\text{subject to} \ \ u_i + v_j \le 1 + I(C_{ij} \ge \epsilon), \forall i, j,$$
$$\mathbf{u} \in \{0,1\}^m, \mathbf{v} \in \{0,1\}^n$$

△Relax to a continous interval and impose sparsity-inducing norm:

$$\max_{\mathbf{u},\mathbf{v}} \ \sum_{ij} C_{ij} u_i v_j - \lambda_u \|\mathbf{u}\|_1 - \lambda_v \|\mathbf{v}\|_1$$
$$\text{subject to} \ \ u_i + v_j \le 1 + I(C_{ij} \ge \epsilon), \forall i, j \ ,$$
$$\mathbf{u} \in [0,1]^m, \mathbf{v} \in [0,1]^n$$

Algorithm 1: Maximal Biclique Finding (MBF)

Input : Bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, where $\mathbf{W}$ is described by the co-occurrence matrix $\mathbf{C}$; parameters $\lambda_u \ge 0$, $\lambda_v \ge 0$, and ϵ.
Output: Maximal biclique indicated by $\mathbf{u}$ and $\mathbf{v}$

```
Initialize v ← rand(n) ∈ [0,1]^n;
while not converged do
    Compute û_i = min{I(C_ij ≥ ε) − v_j}_{j=1}^n;
    Update u_i = min(I(C_{i,:} v ≥ λ_u), 1 + (û_i)_−);
    Compute v̂_j = min{I(C_ij ≥ ε) − u_i}_{i=1}^m;
    Update v_j = min(I(u^⊤ C_{:,j} ≥ λ_v), 1 + (v̂_j)_−);
```

- Scalable easily to large dataset
  - ☑Lower complexity: $\mathcal{O}(m+n)$, while co-clustering [1] costs $\mathcal{O}(mn^2 + n^3)$ due to an SVD
  - ☑Closed-form updates
  - ☑Parallalizable

- Quality measure of discovered bicliques

$$q(\mathcal{B}) = \frac{1}{|\mathcal{B}|} \sum_{ij} C_{ij} u_i v_j$$

- Similarities and differences with ACA [2] and TCD [3]:

△Similarities:
  - ACA, TCD, and MBF discover visually similar shots in an unsupervised manner.

△Differences:
  - ACA is clustering-based method, and by nature consider all shots in its objective.
  - TCD aims to locate one pair of shots at one time.
  - MBF finds a group of shot pairs at once, and ensures each biclique contains only shots that are similar to each other.

[2] F. Zhou, F. De la Torre, and J. K. Hodgins. Hierarchical aligned cluster analysis for temporal clustering of human motion. TPAMI, 35(3):582–596, 2013.
[3] W.-S. Chu, F. Zhou, and F. De la Torre. Unsupervised temporal commonality discovery. In ECCV, 2012.

Query-specific Video Summarization

- Dateset

△Compiled a YouTube dataset using 10 queries from SumMe [4].
△SumMe contains one video/category; thus is not suitable for our purpose.

- Features

△CENTRIST, Denst-SIFT, HSV color moments (mean, std, skew)
△Shot-level representation using bag of temporal words
△PCA-reduced dimension: 254+3840+108 --> 400

- Subject evaluation

△3 judges see the query, and select >10% and <50% shots they think relevant.
△Ground truth are constructed as the shots selected by >1 judges.
△We used mean average precision (mAP) as an evaluation metric.

- Methods

△k-means (baseline), co-clustering (COC), LiveLight (LL) [4], and MBF.

[4] B. Zhao and E. Xing. Quasi real-time summarization for consumer videos. In CVPR, 2014.

Video query	Length	#Vid	#Frm	#Shot
Base jumping	10m54s	5	17960	241
Bike polo	14m08s	5	22490	341
Eiffel Tower	25m47s	7	43729	381
Excavators river xing	10m41s	3	16019	112
Kids playing in leaves	15m40s	6	27972	238
MLB	12m11s	6	21271	201
NFL	13m28s	3	23179	405
Notre Dame Cathedral	11m26s	5	20110	196
Statue of Liberty	10m44s	5	18542	164
Surfing	22m40s	6	34790	483
Total	147m40s	51	246062	2762

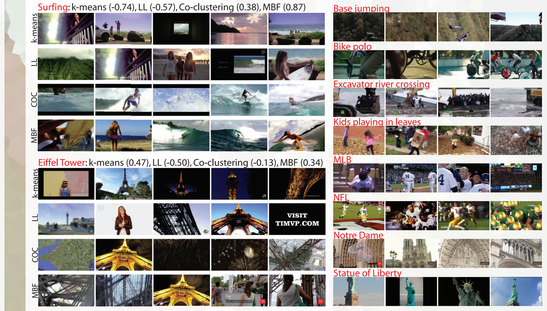
Summaries generated by visual co-occurrence tend to match more closely with human generated summaries!

	Methods	Base*	Bike*	Eiffel*	Excavators*	Kids*	MLB	NFL	Notre Dame*	Statue*	Surfing	*Avg.*
Top 5	*k*-means	0.432	0.427	0.422	0.289	0.791	0.556	0.663	0.392	0.543	0.550	0.507
	LL	0.226	0.305	0.413	0.667	0.744	0.508	0.710	0.568	0.763	0.334	0.524
	COC	0.495	**0.802**	0.580	**0.713**	**0.859**	0.561	**0.762**	**0.803**	0.378	0.668	0.662
	MBF	**0.680**	0.788	**0.596**	0.690	0.798	**0.638**	0.680	0.715	**0.810**	**0.684**	**0.707**
Top 15	*k*-means	0.397	0.369	0.422	0.338	0.772	0.485	0.562	0.442	0.597	0.481	0.487
	LL	0.318	0.459	0.468	0.671	0.710	0.499	0.737	0.592	0.653	0.337	0.545
	COC	0.496	**0.795**	0.561	0.656	0.852	0.503	**0.823**	0.676	0.458	0.586	0.641
	MBF	**0.747**	0.663	**0.562**	**0.674**	**0.859**	**0.755**	0.760	**0.680**	**0.661**	**0.652**	**0.701**

Concept Visualization & Common Motion Discovery

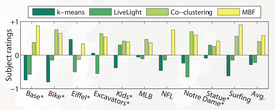
- Concept visualization

- Results



- Subject evaluation

△20 subjects perform an AMT-like webpage, rating good (+1), neutral (0), or bad (-1).



- Common motion discovery

△We used CMU-Mocap dataset with clear-cut labels.
△14 sequences of Subject 86; actions represented by root position, orientation and relative joint angles (30-D).
△We introduce a metric similar to F1-measure.

$$\begin{cases} p(\mathcal{C}_k) &= \frac{1}{|\mathcal{C}_k^{\mathrm{A}}||\mathcal{C}_k^{\mathrm{B}}|} \sum_{i \in \mathcal{C}_k^{\mathrm{A}}, j \in \mathcal{C}_k^{\mathrm{B}}} I(\ell_i = \ell_j), \\ r(\mathcal{C}) &= \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{C}_k^{\mathrm{A}}, j \in \mathcal{C}_k^{\mathrm{B}}} I(\ell_i = \ell_j), \\ F1(\mathcal{C}) &= \frac{2AP(\mathcal{C})r(\mathcal{C})}{AP(\mathcal{C})+r(\mathcal{C})}. \end{cases}$$

- Results