

VISTO: VIsual STOryboard for Web Video Browsing

Joint work: M. Furini, F. Geraci, M. Montangero, and **M. Pellegrini**

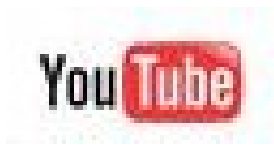


Summary

- Problem, Application, Justification, Scenarios
- Performance Measurements: Time vs Quality
- Our Proposal: VISTO
- Experimental results

Growth of data in Video Format

- YouTube Statistics (Mar 18th, 2008 by Prof Wesch)
- Total videos uploaded as of March 17th 2008: 78.3 Million
- Videos uploaded per day: over 150,000
- Average Video Length: 2 minutes 46.17 seconds
- Time it would take to view all of the material on YouTube (as of March 17th 2008): 412.3 years

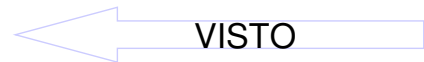


TRECVID 2007: BBC Rushes summaries

- Data: 300 BBC tapes in MPEG-1 format = 50 hours.
- Target produce video skims about 4% of the initial tape-length.
- Basically very little meta data / tagging available
- Quite elaborated evaluation technique.
- Two Baseline algorithms: (1) Random Sample (2) k-means clustering
- Of 22 participants only 3 did better in average “quality” than the two baselines.

Managing Large Video Repositories

- Shot Boundary Detection
- High level feature extraction
- Search, Indexing, Browsing
- Classification
- **Summarization**



Summarization of videos

“A Summary presents a condensed version of some information so that various judgments about the full information can be made using less time and effort”.

- Keyframes (simple or complex layout static storyboards, slideshows)
- Video skims (at fixed or variable speed)
- Mosaic based display (merging several frames into one)
- Exploit analytic capabilities of human vision
- etc..

Selection vs display

Selecting the keyframes (scenes) to be used in a storyboard (skim) is a largely and independent and decoupled task from the display technique.

Quality versus Time

- Quality: be as informative as possible on the video content
- Speed (batch) reduces end-to-end production cycle time.
- Speed (on-line) reduces need for pre-computation and caching, improves personalization of summary generation.

The Visto System in a Nutshell

- Map frames as points in a metric space (HSV format).
- Define a suitable distance function (Generalized Jaccard).
- Estimate the number k of different scenes as a target storyboard size.
- Solve (approximately) the metric k -center problem: split the input points into k groups minimizing the maximum radius of these groups..
- To gain efficiency: exploit temporal/spatial coherence.
- Postprocess to eliminate almost duplicates

Vectorialization of frames

- Many possible different vectorializations of frames in RGB color space: HSV, HSL, CMYK, etc..
- For our experiment we use a 256 entries long HSV vectorialization of an image.
- HSV histograms are supported in MPEG7, thus relatively easy to obtain from (some) video formats.
- Widely used as color space.
- Close to human perception of colors

The Metric Distance Function

Generalized Jaccard Distance:

$$\begin{aligned} a &= (a_1 \cdots a_h) \\ b &= (b_1 \cdots b_h) \\ \rightarrow \rightarrow & \\ GJD(a, b) &= 1 - \frac{\sum_{i=1, h} \min(a_i, b_i)}{\sum_{i=1, h} \max(a_i, b_i)} \end{aligned}$$

When a, b are characteristic vectors of two sets, GJD corresponds to the Jaccard distance.

It is a metric.

FPF-algorithm (from 1985)

Input: A set S of n points.

Output: a set T of k centers.

Initialization:

$T = \emptyset$, Pick any point in S and place it in T .

Set for $p \in S \setminus T$, $\mu(p) = \arg \min_{t \in T} D(t, p)$. (leader election)

Loop:

Let p' be the point p in $S \setminus T$ maximizing $D(p, \mu(p))$,
add p' to T .

Update: $\mu(p) = p'$ if $D(p, p') < D(p, \mu(p))$. (leader update)

Termination: when $|T|=k$.

Speed up heuristics

- Apply FPF to a random sample of the input points.
- Use medoids instead of centers when adding extra points.
- If $F(i+1)$ is very close to $F(i)$ add it to the same cluster (few changes of scene).

Experimental evaluation (1)

Data: 7 Short Videos from the Open Video Project.

Data: 4 long videos (1 cartoon, 1 TV news, 1 TV serial, 1 Talk-show).

Competitors:

- K-means
- Delaunay Clustering (Mundur et al.)
- Open Video Project Summaries. (Marchionini et al.)

Experimental evaluation (2)

Quality evaluation (static case):

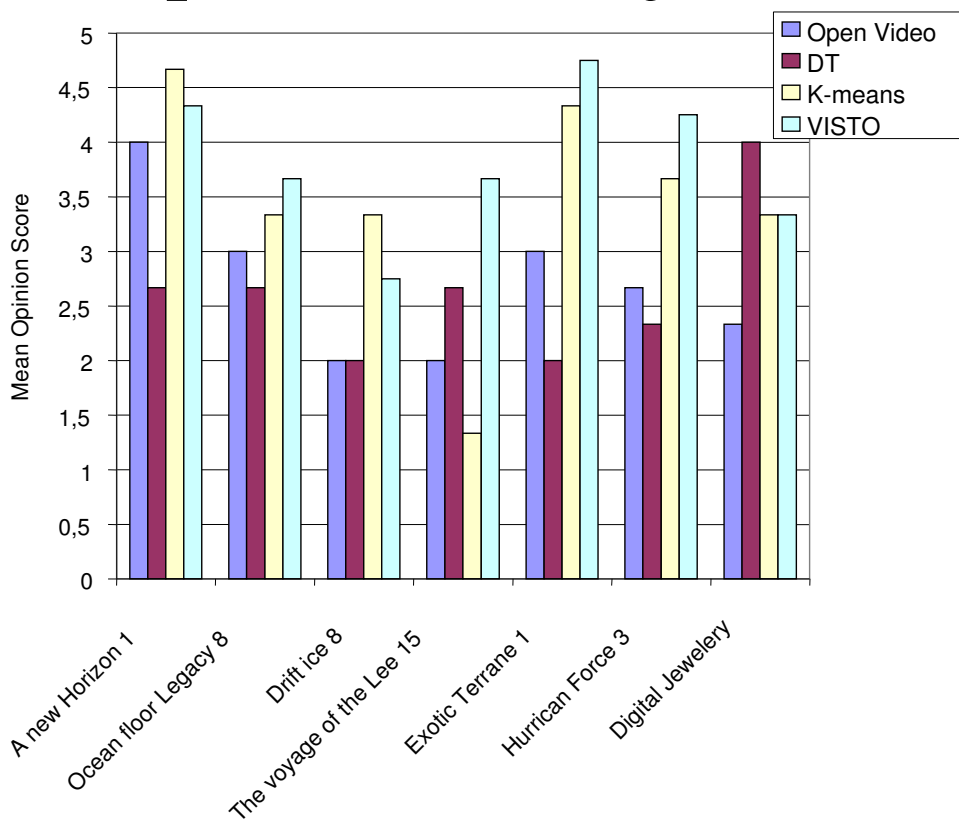
The number of keyframes in the compared summaries is the native one for each algorithm.

Use the same representation and metric when possible.

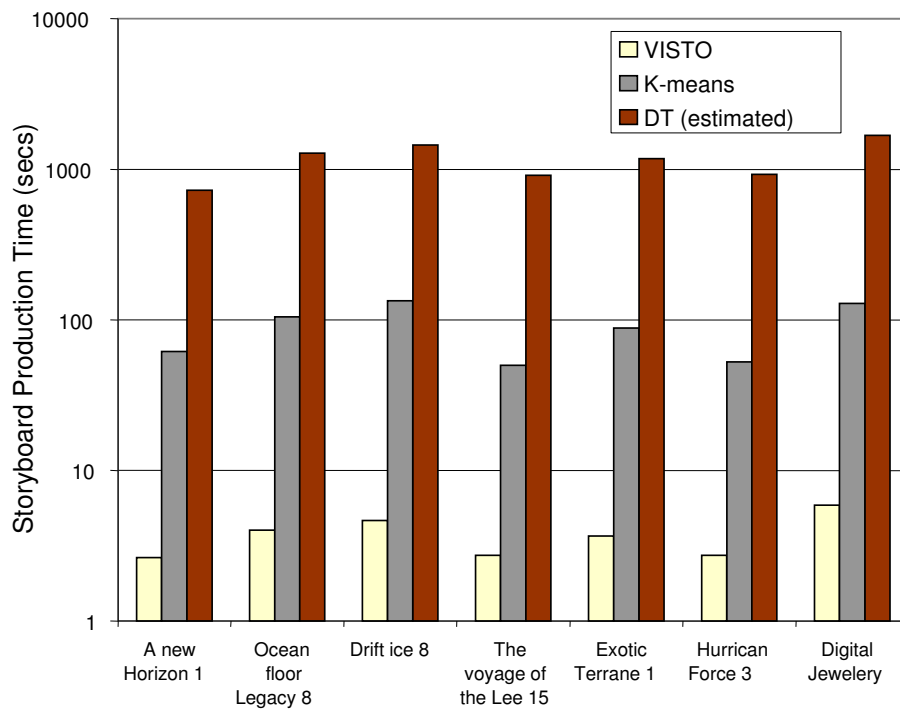
Mean Opinion Score on 20 participants on a scale 1-5 (bad to excellent).

Shown the full video, then the summaries (anonymized) and asked “Is this summary a good representation of the full video?”

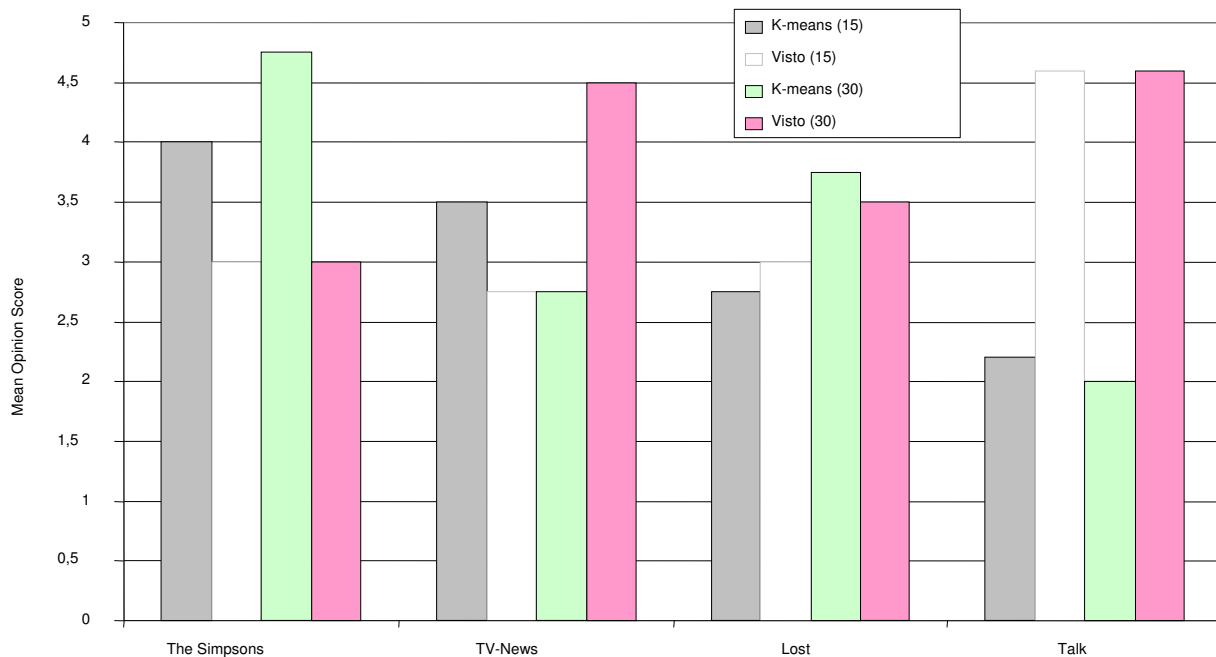
Short “Open Video Project”: Quality



Short “Open Video Project”: Time



Long Videos: Quality



Comparing only VISTO vs K-means with storyboards of 15 and 30 frames.

Video Skims

Unit to be clustered: frames or scenes.

Use also sound to detect scene boundaries

Ground truth. Original video shown twice to 20 volunteers (the second view cut into macro-scenes). Each scene classified in a 0-5 scale (useless to fundamental).

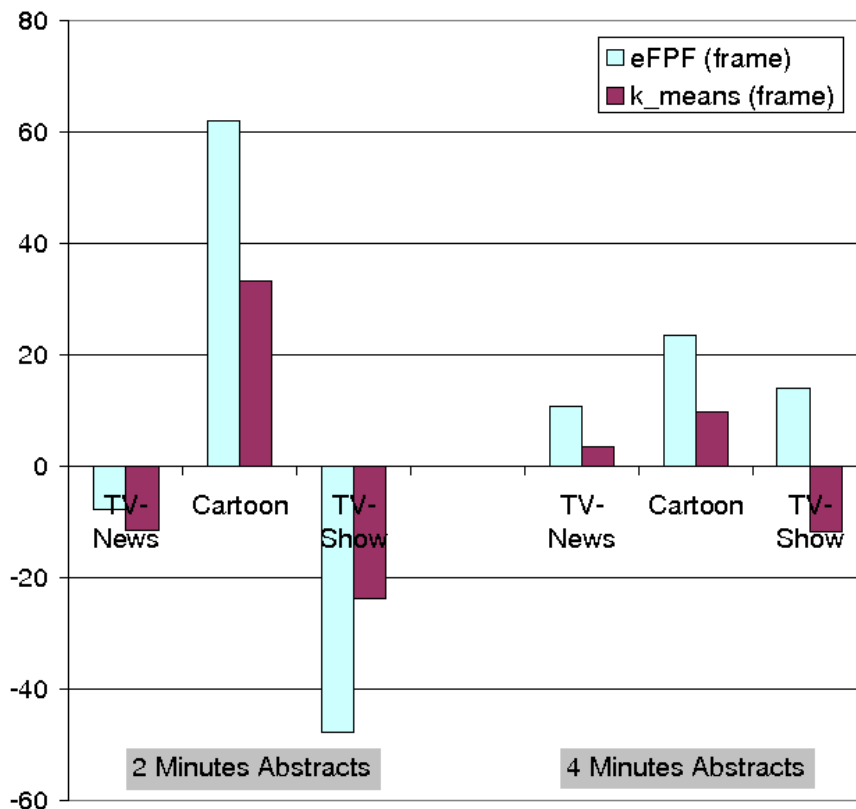
Automatic scoring of scenes in the video skim by the score of the marco-scenes they belong to.

Baseline: random scene selection.

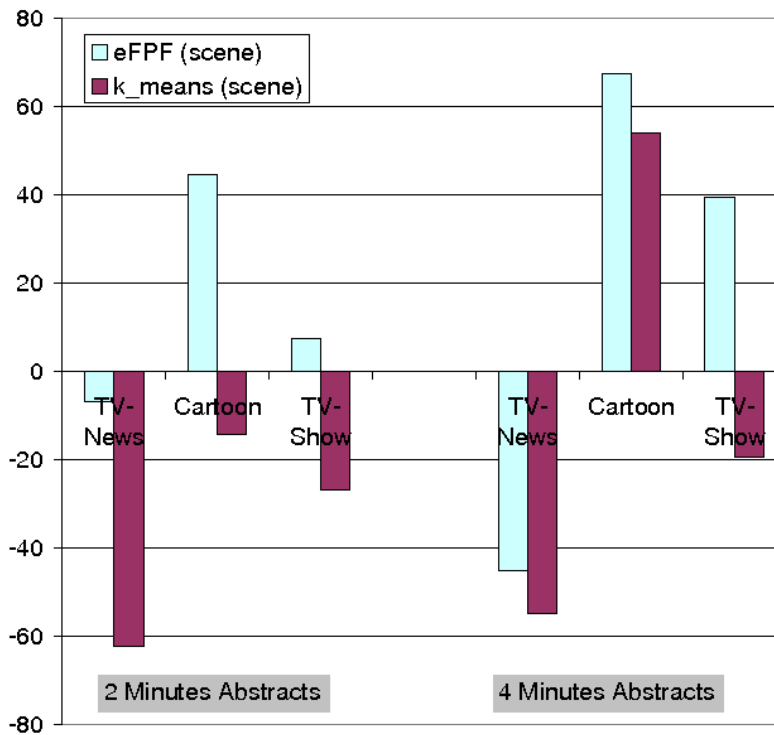
Competitor: k-means

Data: two videos in each category (40 to 15 min long)

Frame-based: Quality



Scene-based: Quality



Conclusion

- It is possible to produce good quality static storyboards on-the-fly (for short videos).
- It is possible to produce good quality static storyboards for long videos in a fraction of their length.
- Preliminary results on video skims are encouraging
- Need for more extended testing (TRECVID style).

References

- M. Furini, F. Geraci, M. Montangelo, and M. Pellegrini: *VISTO: visual storyboard for web video browsing*. ACM CIVR 2007, pp. 635-642.
- M. Furini , F. Geraci, M. Montangelo, and M. Pellegrini: *On Using Clustering Algorithms to Produce Video Abstracts for the Web Scenario*. IEEE CCNC 2008, pp. 1112-1116.