**OBJECTIVES**
- Develop a scalable representation of the bitstream useful for video summarization.
- Efficient generation of the bitstreams.
- Develop a flexible method able to generate summaries from short summaries (few frames) to medium-large summaries in a gradual manner and with reasonable results.

**APPROACH**
- Analysis and generation are decoupled (lightweight generation using bitstream extraction).
- Two criteria for satisfactory summary:
  - Semantic coverage: contains most of the information in the content.
  - Visual smoothness: without annoying visual artifacts if possible.
- Use Group of Pictures (GoP) as basic unit for both analysis and generation.
- Method: order GoPs as in coding order.
- Both video skims and storyboards.

**Why iterative ranking?**
The appropriate summarization technique depends on the target size of the summary:
- Short: mainly semantic coverage.
- Longer: balance both coverage and smoothness.

We tackle both cases separately, but in a gradual manner. Ranking allows to balance both criteria and adapt them as required. The iterative scheme allows to improve progressively the summary.

**RESULTS**
Example: storyboard.
- News12 (from MPEG-7 content set)
- H.264/AVC, GoP length = 8 frames

1. Analysis
   - Keyframe selection
     - Given a constraint (e.g. length) find the cut-off index \( k^* \).
     - Select the first \( k^* \) GoPs and sort them in coding order.
   - Bitstream extraction
     - Simply keep those packets corresponding to the selected GoPs
       - Storyboards: only I frames (slices)
       - Video skims: whole GoPs

2. Generation
   - Formulation of video summarization as excerpt growing (with a novel ranking algorithm).
   - Scalable representation \( \Rightarrow \) "analyze once, generate many"
   - Efficient analysis and generation.

**CONCLUSIONS**
- Flexible algorithm for both storyboard and skim summaries.
- Formulation of video summarization as excerpt growing (with a novel ranking algorithm).
- Scalable representation \( \Rightarrow \) "analyze once, generate many"
- Efficient analysis and generation.

**GENERATION**

**ANALYSIS**

1. GoP level
   - Shot change detection (compressed domain)

2. Shot level
   - Rejection of unsuitable GoPs: those belonging to shot changes and very short shots.
   - Candidate sampling: for each shot, from 1 to 3 representative keyframes are selected
   - Feature extraction of the representative keyframes.

3. Sequence level
   - Clustering (spectral) of the representative keyframes into clusters (group similar shots).
   - GoP ranking: ranks the GoPs iteratively trying to find the best GoP that improves previous summary.

**GoP ranking**

1. Cluster level ranking
   - Mark every cluster as unselected
   - Compute the score for every unselected cluster
     \[ \text{score}(c_i) = (1 - \alpha \cdot \text{duration}(c_i)) \cdot \text{score}(c_i) + \alpha \cdot \text{score}(c_i) \]
   - Select cluster with maximum score, mark as selected and add it to the ranked list (one GoP for storyboards and a segment of GoPs for skims).
   - Repeat until all clusters are selected

2. Shot level ranking
   - Compute the score for every shot, except those completely selected
     \[ \text{score}(s_j) = (1 - \alpha \cdot \text{duration}(s_j)) \cdot \text{score}(s_j) + \alpha \cdot \text{score}(s_j) \]
   - Select shot with maximum score
     - If the shot was not selected before, add a segment of GoPs to the list.
     - If it was already selected, add a new GoP at its boundary (excerpt growing)
   - Repeat until all shots are selected

3. Ranking of discarded GoPs
   - Begin including the GoPs of discarded shots, and then GoPs with shot changes.