Video Google: A Text Retrieval Approach To Object Matching in Videos [Sivic, Zisserman (ICCV 2013)]

This was the first paper which applied the text retrieval techniques to object detection in videos.

The method mostly follows the text retrieval pipeline. Major steps involved are:

1. **Viewpoint invariant description**
    - Two types of viewpoint covariant elliptical regions computed about interest points : shape adapted (SA) and maximally stable (MS)
    - Correspond to corner and blob interest points
    - 128 dimensional SIFT descriptors computed at these regions
2. **Building visual vocabulary**
    - Vector quantize the descriptors using K means : analogue of 'words' in text retrieval
3. **Visual indexing**
    - Every 'document' is represented by a vector of word frequencies
    - tf-idf reweighting
    - inverted index
    - retrieval performance measured by normalized rank of releavant images.
    - 'Stop' word analogy: top 5% and botoom 10% common words are dropped
    - spatial consistency conditions are imposed for matching

Experiments:

164 frames from 48 shots of 19 3D locations of Hollywood movie 'Run Lola Run'.   Rank of 0.0132 reported but since this is the first paper there is no comparision.

Event Retrieval in large video collections with circulant temporal encoding [Revaud et al, CVPR 2013]

- This paper deals with retrieving videos for a specific event, for example Obama's victory speech, i.e. temporal videos are localized over time period and about the same event.
- They also contribute **EVVE** dataset of 13 events for this type of event detection
- The key idea is to jointly encode in a single vector the appearance and temporal information of frames in a video
- Steps:
    - **Frame description:**

        - Preprocess videos to resize to fixed size
        - Densely sampled SIFT description
        - Aggreagate SIFT descriptors into a single vector using MultiVLAD
    - **Circulant temporal aggregation (CTE)**
        - For a pair of videos q = [q_1, q_2, …, q_n] and b = [b_1, b_2, …, b_n] , the inner product between q_i and b_j represents the similarity between frames.
        - Sum of similarities between frames reflects the similarities of the sequences
        - Hence we can represent the similarity between two videos by a circulant matrix whose rows convolution between q and b. Each row represents a different shift
        - This computation can be done efficiently by transforming to the Fourier domain
        - Product quantization (of complex numbers) and tabulating all possible squared distances is done for speedup
        - Higher frequencies are pruned
- Experiments:
    - **Video copy detection:** beats state of art on CCWEB and TRECVID2008 datasets
    - **Event detection:** Comparison with Mean-MultiVLAD (MMV) : average of all frame descriptors for a video with simple dot product for comparing MMVs.
    - **Automatic Video Alignment :** match all possible videos of an event calculating the shift and align all of them to a common timeline by using linear-least squares to prune outliers.

Let
d : MultiVLAD feature vector dimension
n : number of frames in video
N: Number of database videos

- **Computational benefits of going to frequency domain**
    - Query frame descriptors mapped to frequency domain :
    O(d x n log n)
    - Higher frequencies can be pruned retaining only n’ = beta x n : fraction of low frequency feature vectors
    - Product quantization and distance metric optimization by lookup table of distance between ‘p’ centroids, producing p x n’ bytes representation for the video:
    O(256 x p x n’)
    - Due to temporal consistency and self similarity of frames in video the value of comparison score vector s(q,b) are noisy and it peak over is not precisely localized.
        - We can handle this by an additional filtering stage in the Fourier domain
        - Two complementary methods of regularization

        - Use the properties of circulant matrices
        - O(N x p x n’)
    - Map to temporal domain using single inverse FFT: O(N x n’ log n’)

As seen from the above points frequency domain representation enables better “regularized” comparison metric as well as reduced computational time complexity and low memory footprint.