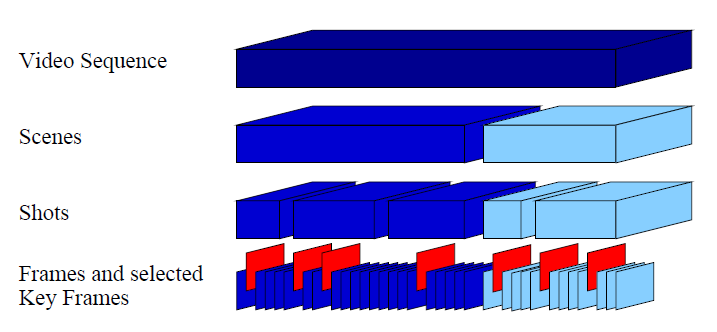
**Rushes Editing**

**Automatic Raw Footage Editing**

# 1. Introduction

There has been brought an attention to multimedia technologies with its rapid development especially in movie industry. The raw footages of the movies are considerably in huge amount and they are unorganized. **Rushes editing**, which enables the computer to edit the film based on the raw footage like a professional film cutter, attracts more and more attention in recent years. The most critical problem of rushes editing is how to generate an effective, efficient, and robust descriptor for the footage content analysis. **Rushes** are the unorganized raw footage with considerable noise and redundancy, which are the source of the usable scene for the final movie [1]. Rushes contain many takes of the same scene redone due to errors. So, they include many frames or sequences of frames that are highly repetitive.



**Figure 1.1: Hierarchical structure within a video sequence**

A video sequence (rushes) has different scenes. Each scene has different shots (retakes). Key frames for each useful shot are selected for indexing to enable users to pick the clips to form a complete video. There are three main problems to be tackled in rushes editing. They are:

* Shot Boundary Detection: segments the video sequence (rushes) into shots
* Junk Frames Removal: filter the irrelevant frames in each shot
* Inter-shot Redundancy Removal: remove the useless retakes in each scene. i.e. select the last retakes in each scene.

## 1.1 Shot Transition

Rushes mainly contains hard-cut transition among shots. The between scenes are most likely switched with junk frames, e.g. color bar, pure color frames. There is also another abnormal case that a Hard-Cut transition is caused by two similar shots.

## 1.2 Junk Frames

The common junk frames of each shot for rushes are color bars frames, pure color frames and the preparation period before the shot take place (camera adjustment, actors getting into right position).

## 1.3 Retakes

Each scene may have many shots (retakes). The retakes contain same contents. To do summary for every scene, it is necessary to select a final shot in every scene as representative clip.

# 2. Methodology

**Figure 2.1 Rushes Editing Framework**

## 2.1 Shot Boundary Detection

Frame difference is a scoring to tell the similarity/difference of two different images. Block based algorithm is chosen as my primary algorithm for scoring frame difference. It is a reference algorithm in Java image cookbook. It is called “Naïve Similarity” [2]. The algorithm is stated as:

1. The two consecutive pictures are normalized into dimension of 300 x 300 pixels using nearest neighbor Interpolation for resampling.
2. Each of the pictures is then divided into 25 blocks. Unlike traditional block based comparison, it does not store the texture and variance of each block. Instead, the color average of the blocks is calculated as B(i,c). Where *i* is a certain block and *c* is the color component of the color model (e.g. RGB, YUV).
3. The *frame difference* between two consecutive pictures is calculated with Euclidean distance as follow:

Decision model is to determine when there should declare a shot boundary based on frame difference scoring. It is usually done to check whether frame difference passes a/different threshold(s). We set a predefined threshold to be 1200.

## 2.2 Junk Frame Removal

There are many cases of junk frames. To deal with different situations, users should be able to set lists of junk frames. These junk frames are then compared with the current frame to check similarity using “Naïve Similarity”. Once the frame difference between current frame and any of the junk frames, passes a threshold Tj which is calculated by Tolerance (%) \* Shot Detection Threshold (T), the current frame is regarded as junk frames. For deciding a proper tolerance, it is more likely under 20% since junk frames would always be different in large scale. Also, Junk segments usually occur at switching shots, from black frame to a very short segment (less than 0.5 seconds) which contains complex color frames. (A shot usually longer than 0.5 seconds including clapper board and actors preparation, it is reasonable to remove a shot for less than 0.5 seconds)

## 2.3 Retakes Removal

Using SIFT (Lowe, David G. (1999)) [3] to match the features of two different frames, it is most likely to track objects and events for every frame. If the feature matching points Pm between two shots attain a predefined threshold Tf , which is calculated as: Number of keyframes selected for each shot ^2 multiplied by expected features for two similar image. The two shots are considered to be in same scene (similar shots). Let FP(*i, k*) be the number of matching points between two frames, where *i* is a frame in a shot and *k* is a frame in another shot. *, where n and s are number of selected keyframes for the two compared shots.* The cluster algorithm is stated as:

1. Mark each shot a sequence number in ascending order
2. Select a shot with least sequence number from the pool and create a cluster for containing it. If there is any, go to step 6.
3. Compare the selected shot with other shots in the pool and then test with condition Pm > Tf . Put the shots which matches with the condition in the same cluster of the selected shot.
4. Select the highest sequence number shot in the cluster and mark it as a representative clip. Remove the cluster. (remove shots in the cluster for further comparison)
5. If there are no shots in the pool, go to Step 2.
6. Finish.

The last shot for a cluster should contain less redundant data than other shots in the same scene. So, it is chosen as the representative clip for a particular scene.

# 3. Further Improvement

An adaptive threshold should be chosen for shot boundary detection as there are different situations such as fast camera movement, which may lead to false alarm. Also, Junk frames of preparation before shot begins, and after director signals to end the shot should be removed. Inter-shot Redundancy Removal is to be reevaluated to reduce the side effect of the tradeoff between efficiency and effectiveness.

# 4. Conclusion

This project investigates the rushes editing. An application prototype is built to implement its techniques and ideas. The thesis of rushes editing of this work is presented in three areas:

* Shot Boundary Detection: detection of shot transition and cut points are declared to separate the video into different shots.
* Junk Frame Removal: pure color frames, color bar frames and other junk frames defined by users are removed from each shot. This is done via comparing similarity between normal frames and the junk frames defined.
* Inter-shot Redundancy Removal: shots are clustered using features matching and then each cluster selects the last shot as the representative shot for each scene.

# 5. References

[1] Yang Liu, Yan Liu, and Keith C.C. Cham, Dimensionality reduction for Heterogeneous Dataset in Rushes Editing

[2] http://www.lac.inpe.br/JIPCookbook/6050-howto-compareimages.jsp

[3] Lowe, David G. (1999). "Object recognition from local scale-invariant features". Proceedings of the International Conference on Computer Vision. 2. pp. 1150–1157.doi:10.1109/ICCV.1999.790410.