

# [Video shot boundary detection based on color histogram research paper sample](https://assignbuster.com/video-shot-boundary-detection-based-on-color-histogram-research-paper-sample/)

[](https://assignbuster.com/)[Sociology](https://assignbuster.com/essay-subjects/sociology/), [Social Issues](https://assignbuster.com/essay-subjects/sociology/social-issues/)

## Abstract

This paper examines the concept of video shot detection. It examines the use of color histogram differences for shot detection. This method helps to differentiate abrupt and smooth shot boundaries. It involves development of an algorithm that extract abrupt shot discontinuities by analysing the differences in the colour histograms. Histograms colour distributions of an image and the shot boundary effectiveness solely depend on measured similarities between consecutive frames.   
1. Introduction   
Image analysis and processing is an important part of creation of videos. It utilizes mathematical techniques to analyze different sections of an image. Shot boundary detection is a technique of taking two subsequent frames of a video or a photograph taken by the same camera and measuring their differences. This method bases its focus in detecting visual discontinuities along a time domain. These differences are discriminated using different methods. In this case, the discussion will point out on the technique of using the color histogram. This method compares the number of occurrences for each particular color between subsequent frames.   
2. Colour histogram AND SHOT BOUNDARY dETECTION   
2. 1 Introduction   
A color histogram is composed of a range of colors within a range. All colors within this range have a distinct quantization level. Therefore, while looking for the differences the color histogram focuses on an algorithm that brings out differences in colors depending on their quantization levels. This is mainly because the algorithm is able to detect the differences in quantization levels.   
2. 2 Using the color histogram method   
3. Detecting Video shot boundary   
3. 1 Computing the video shot boundary   
In order to understand how to detect the video shot boundary, the color histogram computes gray level of two images in consecutive frames. The bin-wise difference is analyzed and the results are used to deduce the differences between frames.   
This difference is calculated as sum of absolute differences between bin values. The video shot boundary detection method will therefore depend on the differences between colour histograms. A peak on the histogram will indicate a large discontinuity between histograms and this means there is an abrupt transition on the video. For gradual transitions fades and smooth’s appear indicating a constant transition between frames.   
3. 2 Cut Boundary Detection   
After computing the difference in bin values, cut boundary detection is the next critical activity. This is used to eliminate dissolve and fade in the image. Here the signal is smoothened. This is important since it helps to eliminate errors that may be as a result in minute differences in colour histograms. This process helps deal with distortions caused by object or camera movements.   
3. 3 Fade and dissolve detection   
Finally, the fade and dissolve boundary detection is carried out and helps in detection of smooth shot boundaries. The aim of this detection is to be able to detect within a video susceptible frames. When the susceptible frames are detected, the next important thing to consider is to distinguish actual effects in the video as opposed to effects that may result to movement of the camera. Once the real effects have been detected, it is possible to carry out a fade and dissolve detection.   
4. Conclusion   
In conclusion, this method is the most effective method in detecting shot boundaries by comparing pixels in different frames of a video scene using a colour histogram.   
5. Reference   
[1] J. Mas, and G. Fernandes “ Video Shot Boundary Detection Based on Color Histogram,” RECVidWorkshop, Gaithersburg, Maryland USA, pp. 1 – 11, 2003.