Method-1(Report COP290)

Sourav(2019CS10404),Harikesh(2019CS10355)

March 2021

1 Introduction

In implementation of this method we have calculated the queue density and dynamic density using the **absdiff** function.
We also use some **Matplotlib** and **numpy** for plotting the graph and finding **Utility**.
Baseline:
Baseline for this method is the code used in Assignment1 part2 i.e. resolution of frame used (410*1000).

2 Analysis

In this method we use X*Y as where X*Y is the values of resolution to which the frame is reduced.
We have used absolute error as a metric for measuring utility.

$$Runtime = totaltimetakentorunthecode$$

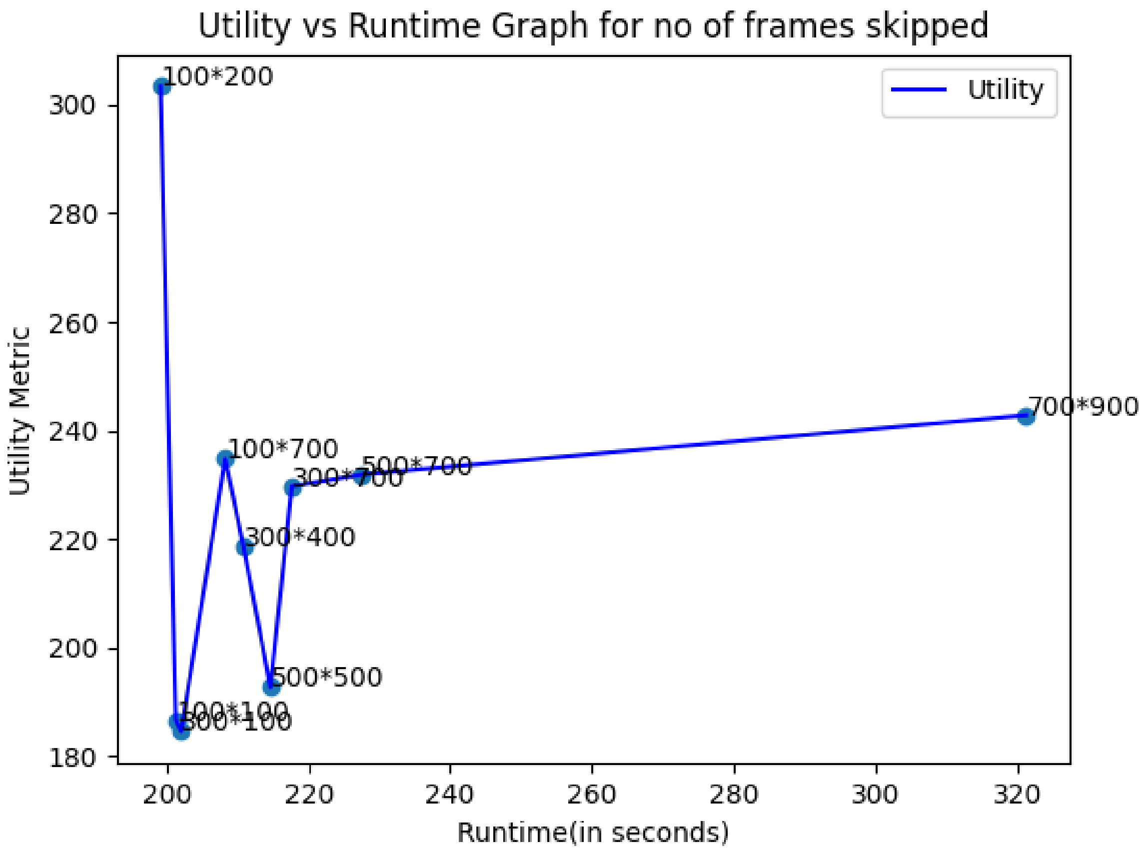
$$queuedensitydeviation = \sum_{allframes} |BaselineDensity - NewDensity|$$

$$dynamicdensitydeviation = \sum_{allframes} |BaselineDensity - NewDensity|$$

$$AbsoluteError = \frac{queuedensitydeviation + dynamicdensitydeviation}{totalframes}$$

$$UtilityMetric = \frac{1}{AbsoluteError}$$

2.1 Utility vs Runtime curve for parameter resolution of the video changes



The graph shown above is the graph for different values of X*Y with their calculated Utility and Runtime values and plot the curve. where X-Axis contains the Runtime for every X*Y and Y-Axis contains the Utility Metric for every x.

As size of the image is less then the time taken for processing is less. this implies that the runtime is less then the error or deviation in high so if deviation is high the utility is decreases.