

Bitrate Targeting Tools

Quick-start Guide

Steve Campbell
Jing Wang
Xiang Yu
Research In Motion Ltd.

Introduction

targetBitrates.sh is a shell script that runs the encoder many times while trying to obtain a specific set of bitrates. It does this by adjusting the Lambda-modifiers that are passed to the encoder.

encodeCommand.sh is a shell script that is invoked by targetBitrates.sh that runs the encoder.

targetBitrates.sh will typically invoke encodeCommand.sh many times during one run. targetBitrates.sh also makes use of two executables: extractBitrates.exe and guessLambdaModifier.exe. It is designed to run in a Bash shell.

Preparation

- Build extractBitrates.exe and guessLambdaModifiers.exe. To do this, execute this command in the folder that contains the source code:

```
make
```
- After building, ensure that these files are all in the same directory:
 - targetBitrates.sh
 - encode.shl
 - encodeCommand.sh
 - extractBitrates.exe
 - guessLambdaModifiers.exe

Run targetBitrates.sh

If you run targetBitrates.sh with no arguments, it will output the usage notes. The usage notes for encodeCommand.sh may also be useful.

Here is an example of a typical set of arguments for targetBitrates.sh

```
./targetBitrates.sh -q 22 -o "~/myOutputDirectory/" -ci ldHE BQSquare_416x240_60 -tb "5000 34241 6541" -ca '-e ~/bin/TAppEncoder.exe -cd ~/cfg/'
```

This runs targetBitrates.sh for QP 22, for configuration low-delay high-efficiency, and for sequence BQSquare_416x240_60. The output will be placed in ~/myOutputDirectory. The target bitrates are specified as "5000 34241 6541". The encoder is located at ~/bin/TAppEncoder.exe and the configuration files are in ~/cfg.

Parameters

The usage of targetBitrates.sh is as follows:

```
./targetBitrates.sh [-rm] -ci configurationIdentifier -q q -tb targetBitrates [-il  
initialLambdaModifiers] [-ca encodeCommandArgs] [-ea extraArguments] -o outputDirectory  
inputName
```

- -rm specifies resume-mode which allows the user to resume an execution that was interrupted before completion.
- configurationIdentifier specifies the configuration (ldHE, ldLC, raHE, raLC, inHE, or inLC).
- q is the QP value (22, 27, 32, or 37).
- targetBitrates is the target bitrates for each temporal layer separated by spaces. For example: "3445 3473 etc...".
- initialLambdaModifiers is the Lambda-modifiers to use for the first guess. For example: "-LM0 1e0 -LM1 0.98 etc...".
- encodeCommandArgs is the extra arguments to be passed to encodeCommand.sh. The common arguments that are available to both ./targetBitrates.sh and encodeCommand.sh should not be passed through this argument. For example, don't pass -q here because it is an option of ./targetBitrates.sh. -e and (-cp or -cd) must be passed through this argument. For example, "-ca '-e ~/bin/encode.exe -cd ~/cfg/'".
- extraArguments specifies extra arguments to be passed directly to the encoder (not to encodeCommand.sh).
- outputDirectory is the directory that will contain the output logs, YUV, and bin.
- inputName is the name of the input sequence. Must be one of the following:
 - NebutaFestival_2560x1600_60_10bit_crop
 - SteamLocomotiveTrain_2560x1600_60_10bit_crop
 - Traffic_2560x1600_30_crop
 - PeopleOnStreet_2560x1600_30_crop
 - BQTerrace_1920x1080_60
 - BasketballDrive_1920x1080_50
 - Cactus_1920x1080_50
 - Kimono1_1920x1080_24
 - ParkScene_1920x1080_24
 - vidyo1_720p_60
 - vidyo3_720p_60
 - vidyo4_720p_60
 - RaceHorses_832x480_30
 - BQMall_832x480_60
 - PartyScene_832x480_50

- BasketballDrill_832x480_50
- RaceHorses_416x240_30
- BQSquare_416x240_60
- BlowingBubbles_416x240_50
- BasketballPass_416x240_50
- BasketballDrillText_832x480_50
- Chinaspeed_1024x768_30
- SlideEditing_1280x720_30
- SlideShow_1280x720_20

Output

The output files are all placed in the output directory that is specified by the user. These files include:

- The YUV file. There is only one YUV file as it is repeatedly overwritten with each invocation of the encoder.
- The BIN file. There is only one BIN file as it is repeatedly overwritten with each invocation of the encoder.
- The `_meta.log` file. This file contains a log for the entire bitrate targeting process. Each line represents one invocation of the encoder and includes:
 - The Lambda-modifiers that were used
 - The bitrates that were obtained
 - The percentage above or below the target bitrates of the obtained bitrates
 - Whether or not this encoding was accepted
- The `_final.log` file. This is the log from the final invocation of the encoder. The bitrates obtained from this log were accepted.
- The `_dep##.log` files. These are the log files from all of the invocations of the encoder except for the final invocation. These logs are labeled as “deprecated” because the bitrates obtained from these logs were not accepted.